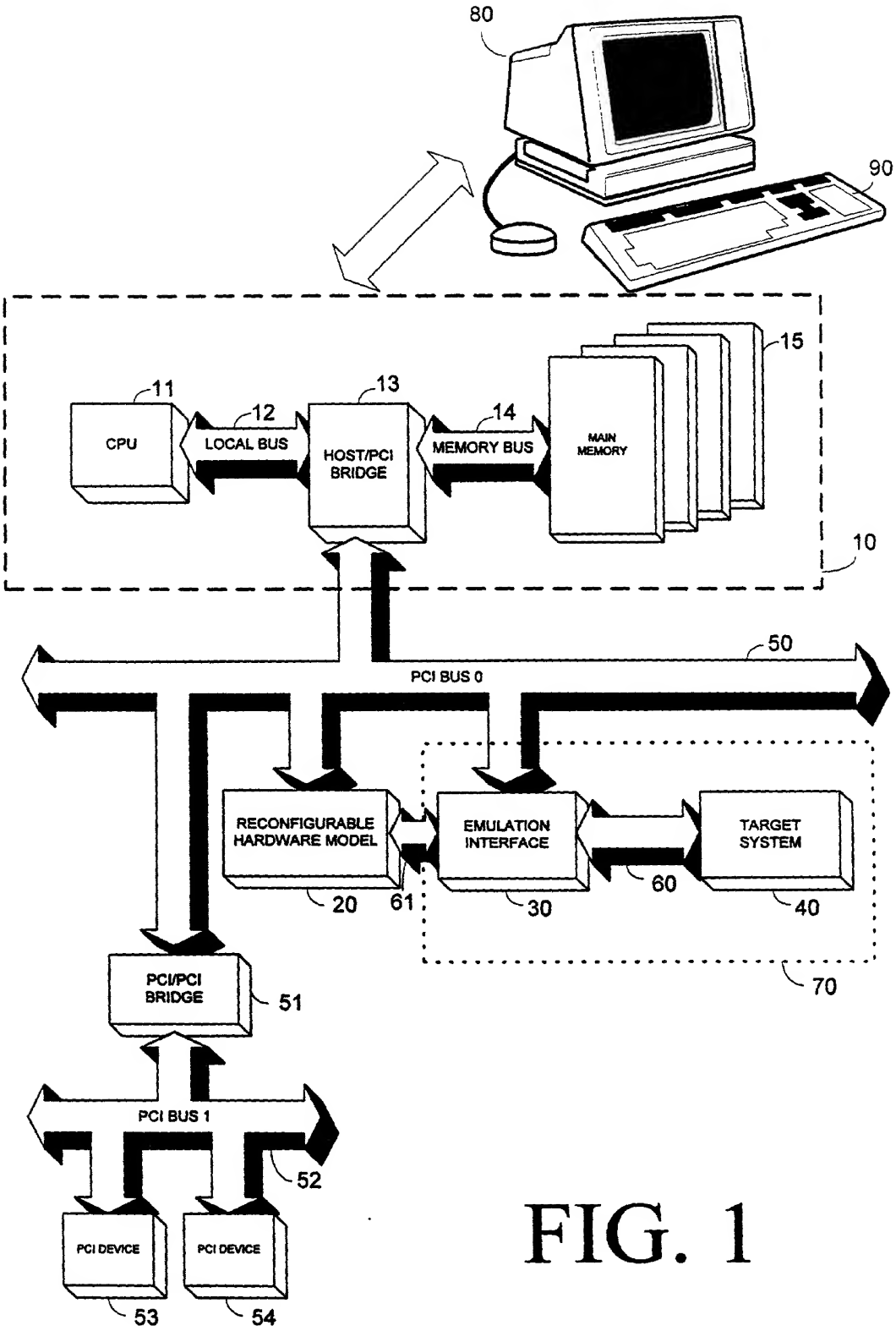

80
90
15
11
12
13
14
CPU
LOCAL BUS
HOST/PCI BRIDGE
MEMORY BUS
MAIN MEMORY
10
50
PCI BUS 0
RECONFIGURABLE HARDWARE MODEL
EMULATION INTERFACE
TARGET SYSTEM
20
61
30
60
40
70
PCI/PCI BRIDGE
51
PCI BUS 1
52
PCI DEVICE
PCI DEVICE
53
54

FIG. 1

USAGE

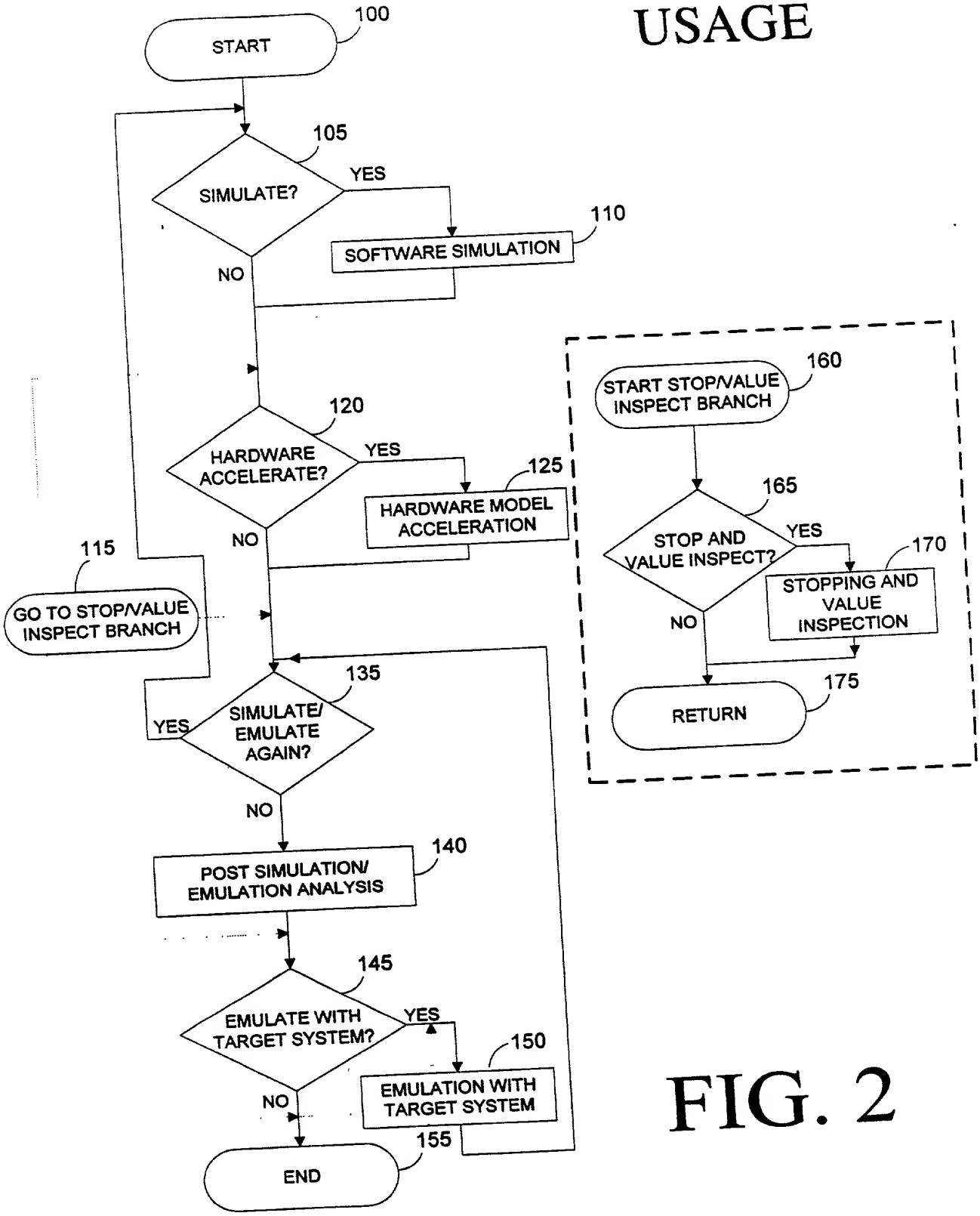

START
100
SIMULATE?
105
YES
SOFTWARE SIMULATION
110
NO
HARDWARE ACCELERATE?
120
YES
HARDWARE MODEL ACCELERATION
125
NO
GO TO STOP/VALUE INSPECT BRANCH
115
SIMULATE/ EMULATE AGAIN?
135
YES
NO
POST SIMULATION/ EMULATION ANALYSIS
140
EMULATE WITH TARGET SYSTEM?
145
YES
EMULATION WITH TARGET SYSTEM
150
NO
END
155
START STOP/VALUE INSPECT BRANCH
160
STOP AND VALUE INSPECT?
165
YES
STOPPING AND VALUE INSPECTION
170
NO
RETURN
175

FIG. 2

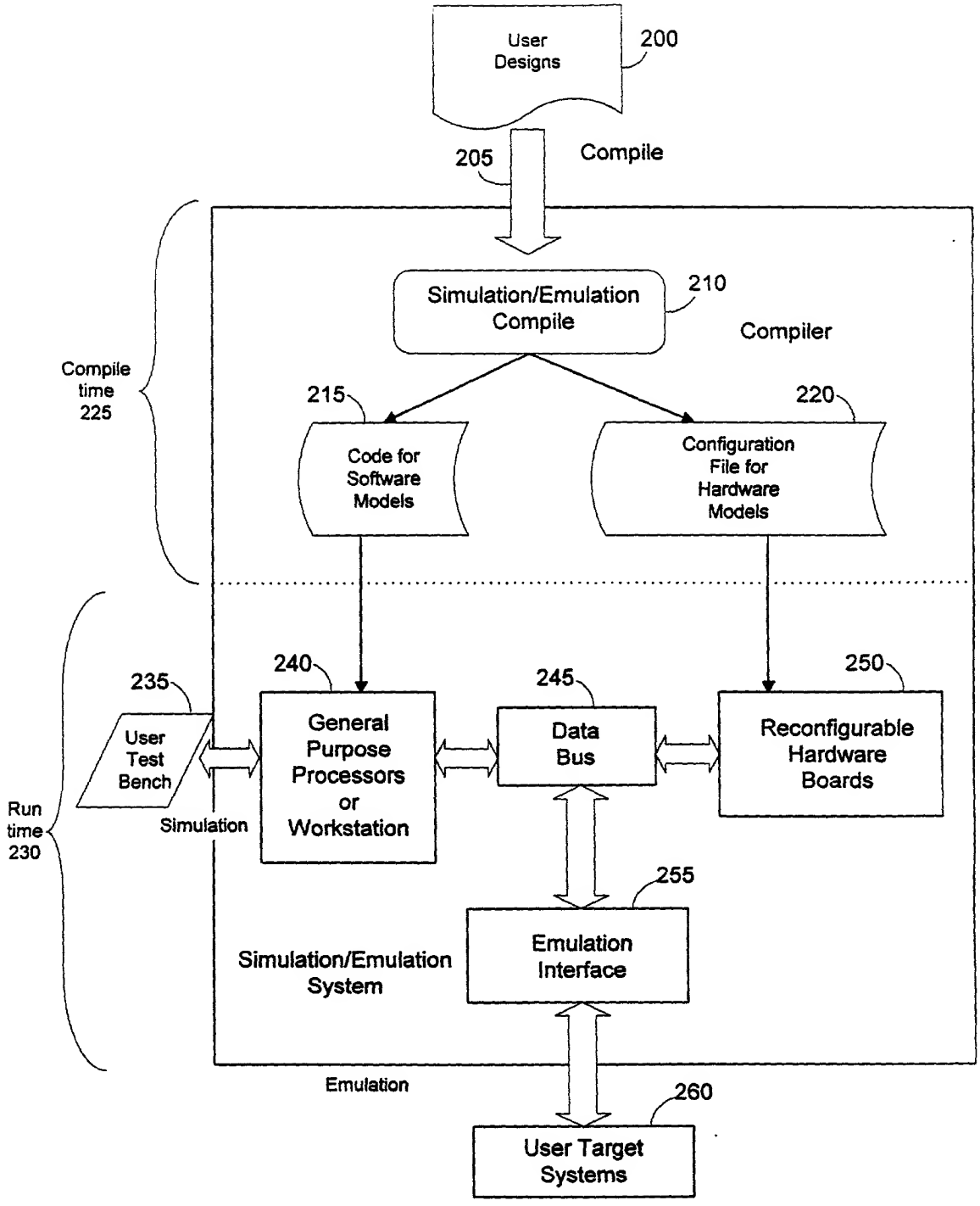

User Designs
200
205
Compile
Simulation/Emulation Compile
210
Compiler
Compile time 225
215
220
Code for Software Models
Configuration File for Hardware Models
235
240
245
250
User Test Bench
General Purpose Processors or Workstation
Data Bus
Reconfigurable Hardware Boards
Run time 230
Simulation
255
Simulation/Emulation System
Emulation Interface
Emulation
260
User Target Systems

FIG. 3

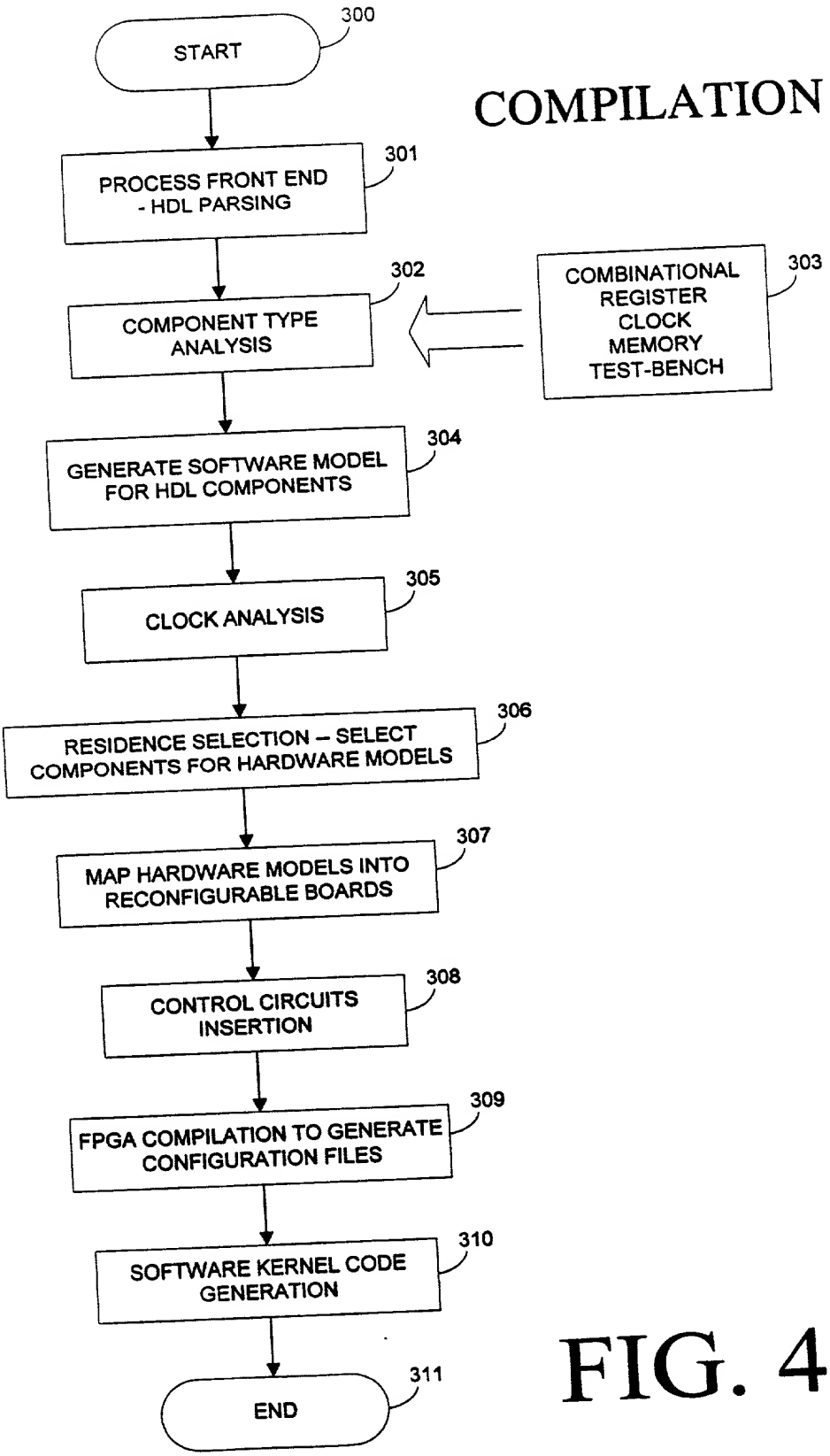

COMPILATION
300
START
301
PROCESS FRONT END
- HDL PARSING
302
COMPONENT TYPE
ANALYSIS
303
COMBINATIONAL
REGISTER
CLOCK
MEMORY
TEST-BENCH
304
GENERATE SOFTWARE MODEL
FOR HDL COMPONENTS
305
CLOCK ANALYSIS
306
RESIDENCE SELECTION – SELECT
COMPONENTS FOR HARDWARE MODELS
307
MAP HARDWARE MODELS INTO
RECONFIGURABLE BOARDS
308
CONTROL CIRCUITS
INSERTION
309
FPGA COMPILATION TO GENERATE
CONFIGURATION FILES
310
SOFTWARE KERNEL CODE
GENERATION
311
END

FIG. 4

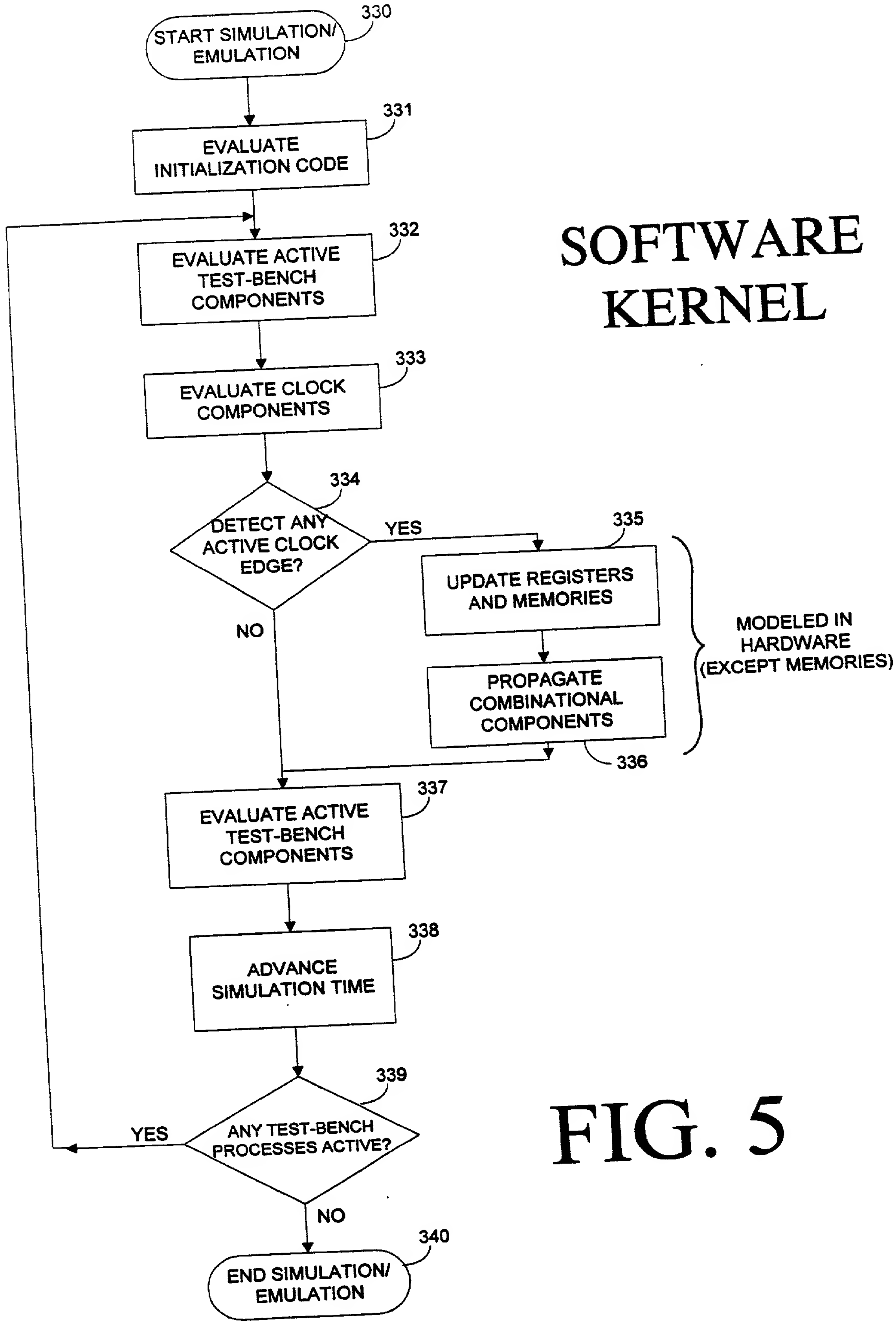

SOFTWARE KERNEL
START SIMULATION/ EMULATION
330
EVALUATE INITIALIZATION CODE
331
EVALUATE ACTIVE TEST-BENCH COMPONENTS
332
EVALUATE CLOCK COMPONENTS
333
DETECT ANY ACTIVE CLOCK EDGE?
334
YES
NO
UPDATE REGISTERS AND MEMORIES
335
PROPAGATE COMBINATIONAL COMPONENTS
336
MODELED IN HARDWARE (EXCEPT MEMORIES)
EVALUATE ACTIVE TEST-BENCH COMPONENTS
337
ADVANCE SIMULATION TIME
338
ANY TEST-BENCH PROCESSES ACTIVE?
339
YES
NO
END SIMULATION/ EMULATION
340

FIG. 5

MAPPING HARDWARE MODELS TO RECONFIGURABLE BOARDS

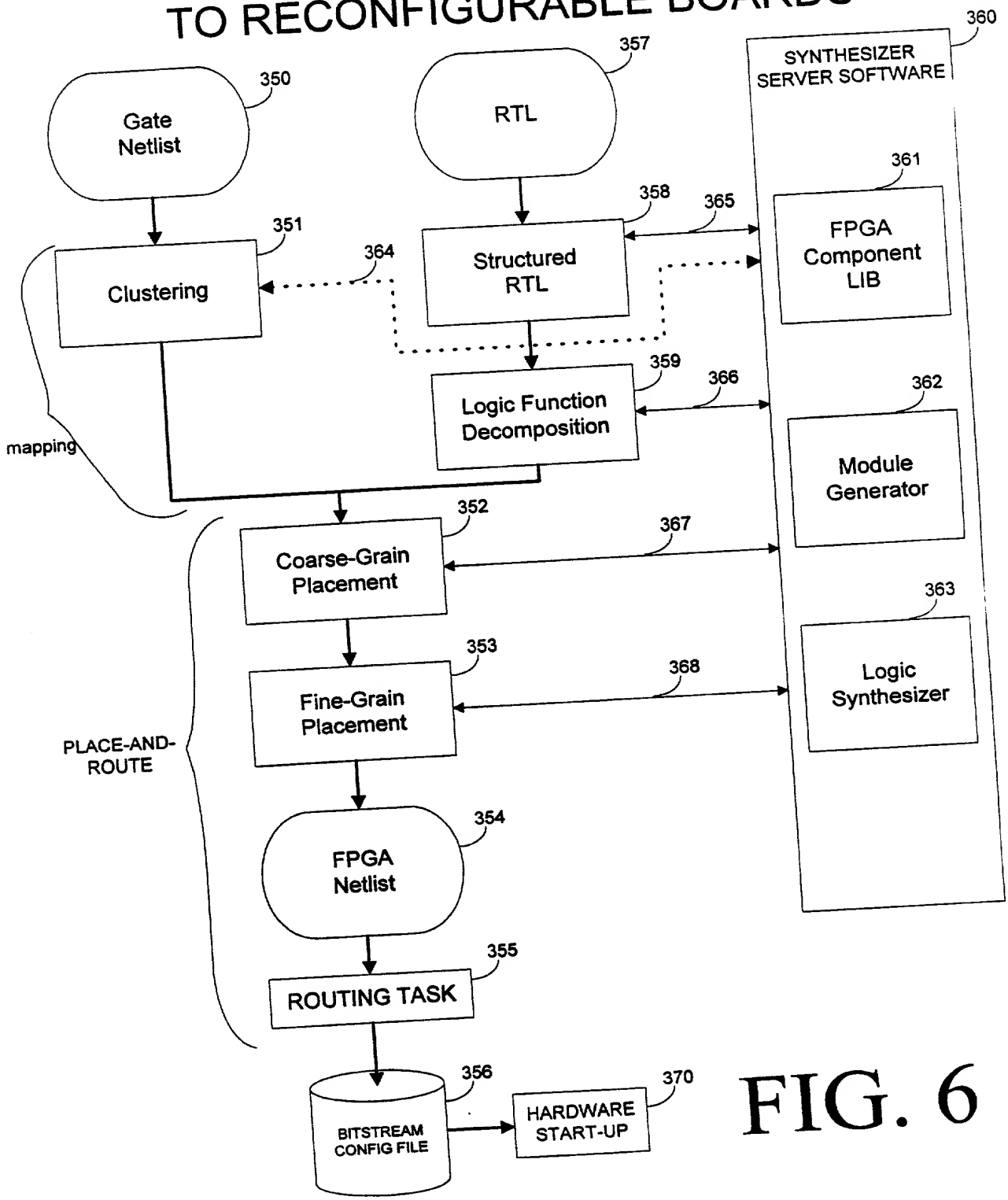


FIG. 6

	F11	F12	F13	F14	F21	F22	F23	F24	F31	F32	F33	F34	F41	F42	F43	F44
F11	1	1	1	1	1	0	0	0	1	0	0	0	1	0	0	0
F12	1	1	1	1	0	1	0	0	0	1	0	0	0	1	0	0
F13	1	1	1	1	0	0	1	0	0	0	1	0	0	0	1	0
F14	1	1	1	1	0	0	0	1	0	0	0	1	0	0	0	1
F21	0	0	0	0	1	1	1	1	1	0	0	0	1	0	0	0
F22	1	1	0	0	1	1	1	1	0	1	0	0	0	1	0	0
F23	0	0	1	0	1	1	1	1	0	0	1	0	0	0	1	0
F24	0	0	0	1	1	1	1	1	0	0	0	1	0	0	0	1
F31	0	0	0	0	1	0	0	0	1	1	1	1	1	0	0	0
F32	1	1	0	0	0	1	0	0	1	1	1	1	0	1	0	0
F33	0	0	1	0	0	0	1	0	1	1	1	1	0	0	1	0
F34	0	0	0	1	0	0	0	1	1	1	1	1	0	0	0	1
F41	0	0	0	0	1	0	0	0	1	0	0	0	1	1	1	1
F42	1	1	0	0	0	1	0	0	0	1	0	0	1	1	1	1
F43	0	0	1	0	0	0	1	0	0	0	1	0	1	1	1	1
F44	0	0	0	1	0	0	0	1	0	0	0	1	1	1	1	1

FIG. 7

FPGA INTERCONNECTION

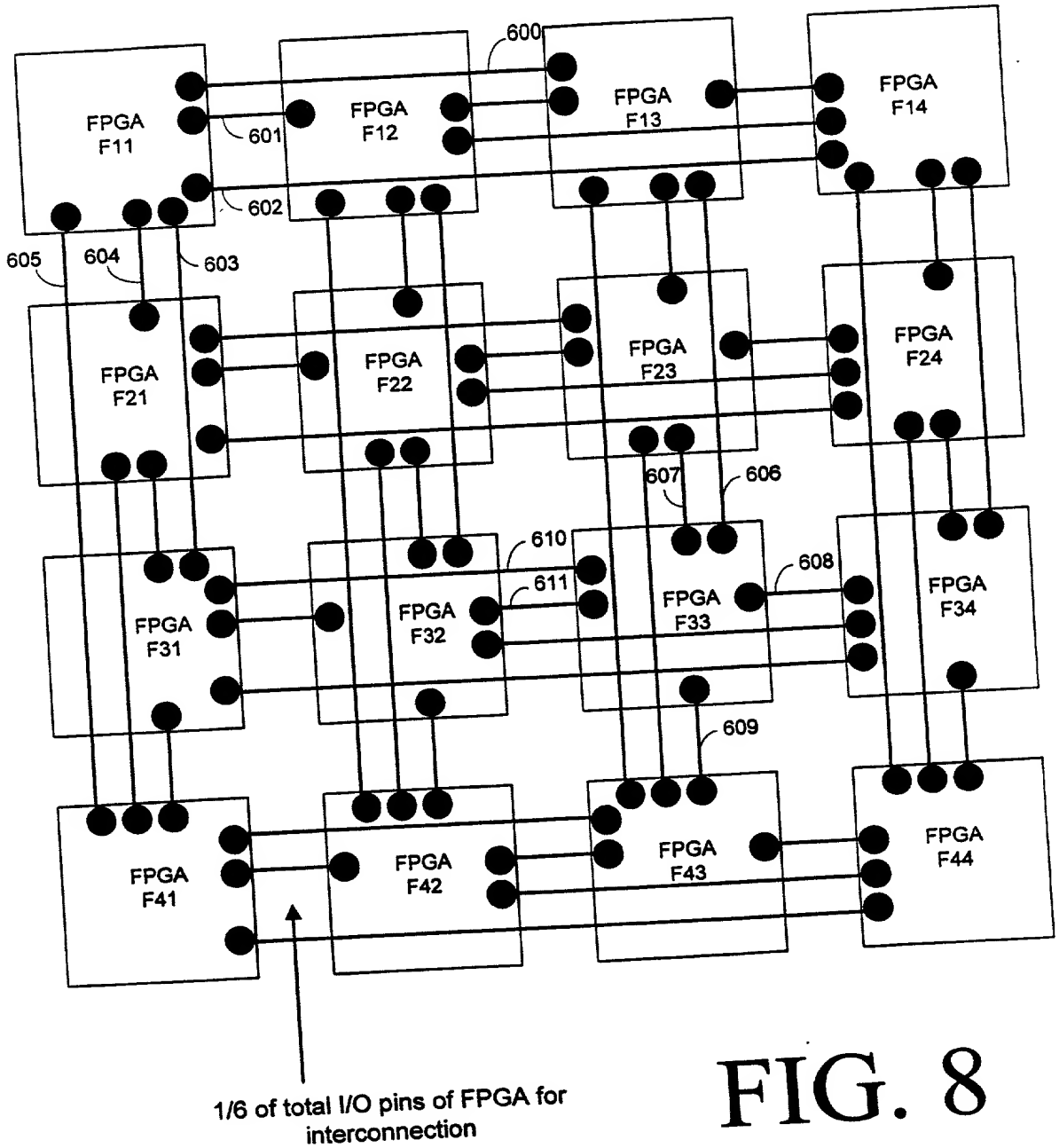
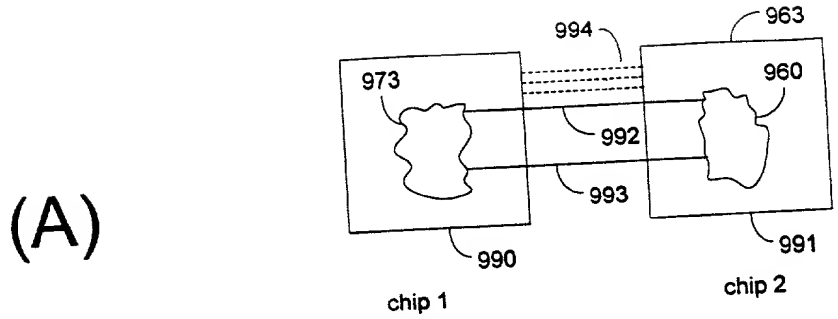
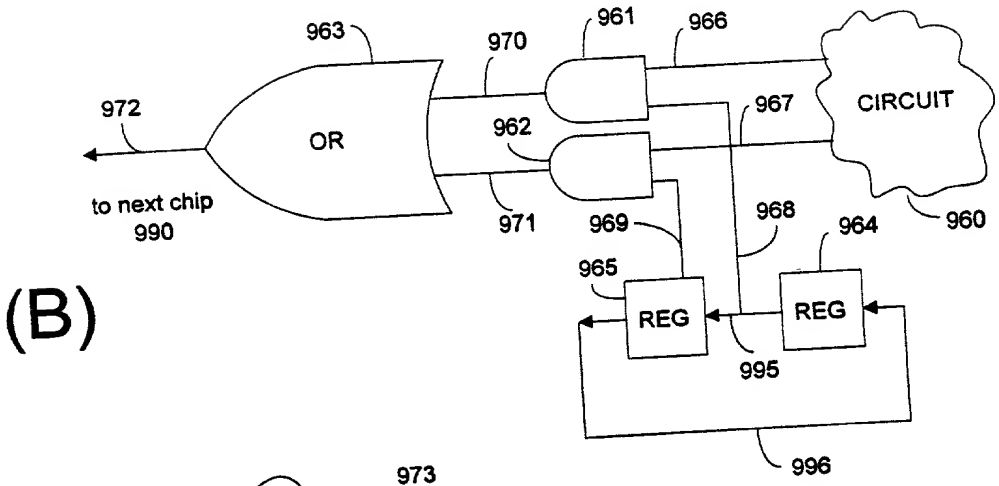


FIG. 8


(A)
994
963
973
960
992
993
990
991
chip 1
chip 2


(B)
963
970
961
966
972
OR
962
967
CIRCUIT
to next chip
990
971
969
968
964
960
965
REG
REG
995
996

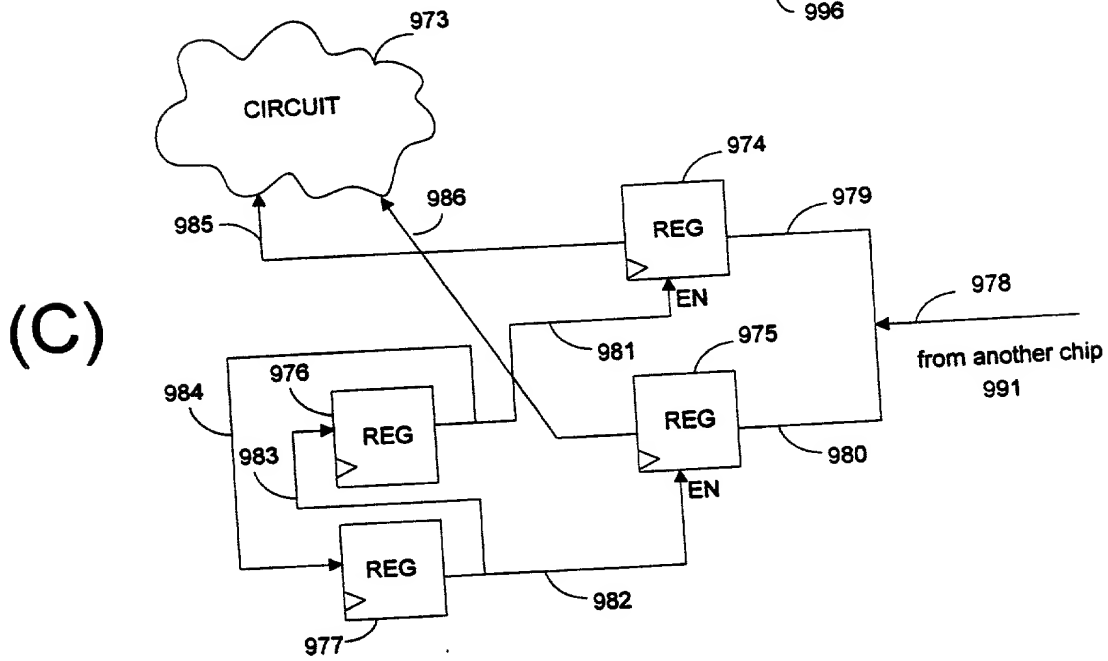

(C)
973
CIRCUIT
974
986
979
985
REG
EN
978
981
975
976
984
REG
REG
from another chip
991
983
980
EN
REG
982
977

FIG. 9

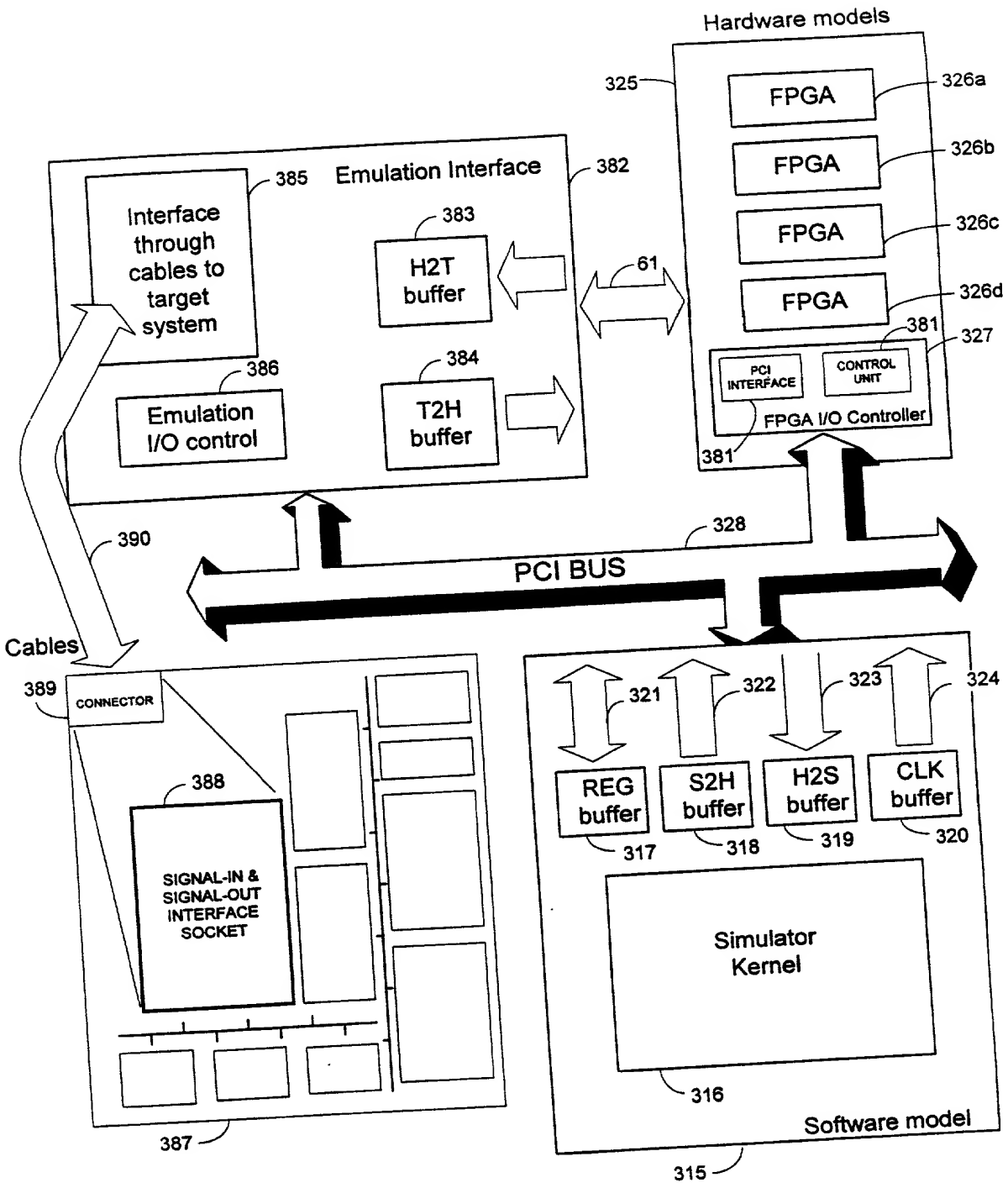

Hardware models
325
FPGA
326a
FPGA
326b
FPGA
326c
FPGA
326d
381
327
PCI INTERFACE
CONTROL UNIT
FPGA I/O Controller
381
Emulation Interface
382
385
Interface through cables to target system
383
H2T buffer
61
386
Emulation I/O control
384
T2H buffer
390
328
PCI BUS
Cables
389
CONNECTOR
388
SIGNAL-IN & SIGNAL-OUT INTERFACE SOCKET
387
321
322
323
324
REG buffer
S2H buffer
H2S buffer
CLK buffer
317
318
319
320
Simulator Kernel
316
Software model
315

FIG. 10

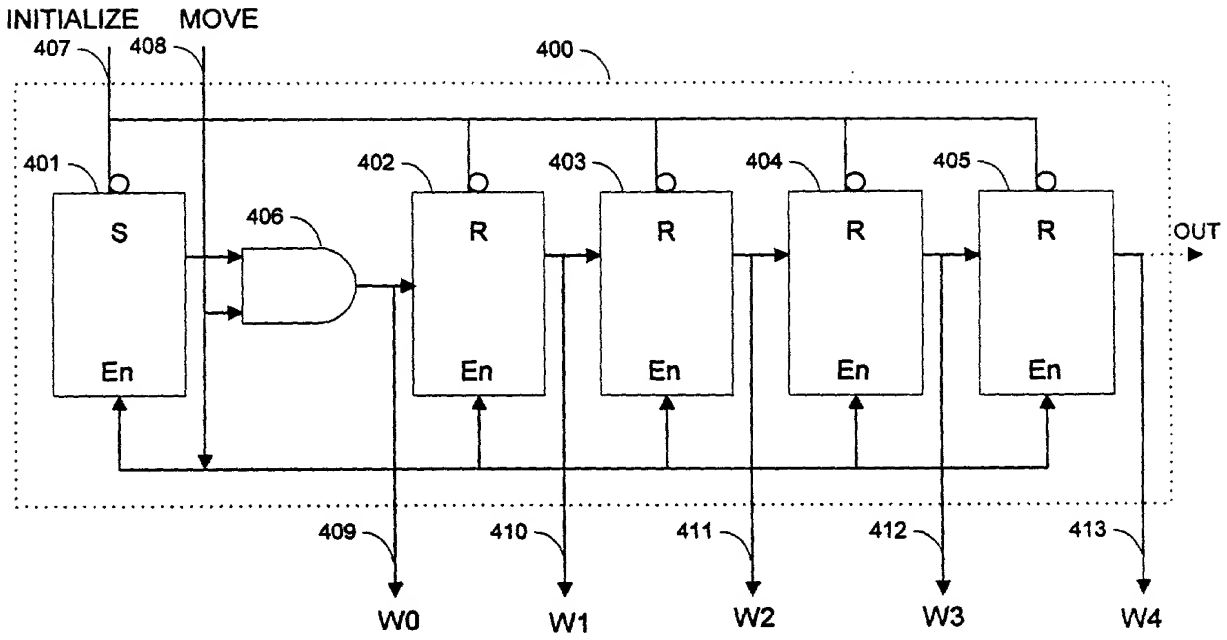

INITIALIZE
MOVE
407
408
400
401
402
403
404
405
S
En
406
R
En
R
En
R
En
R
En
OUT
409
410
411
412
413
W0
W1
W2
W3
W4

FIG. 11

ADDRESS POINTER INITIALIZATION

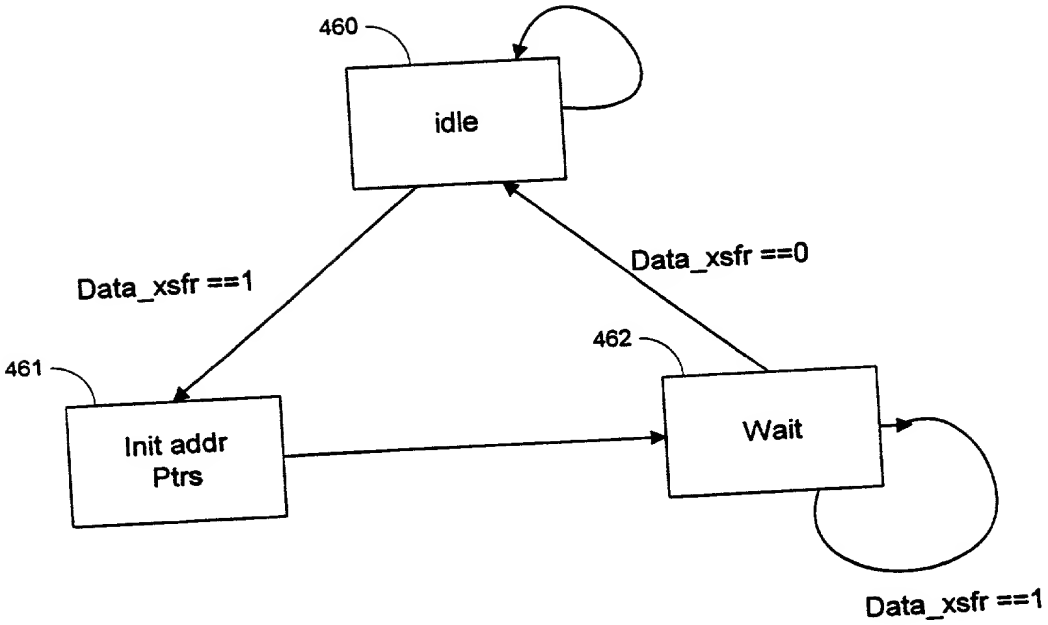


FIG. 12

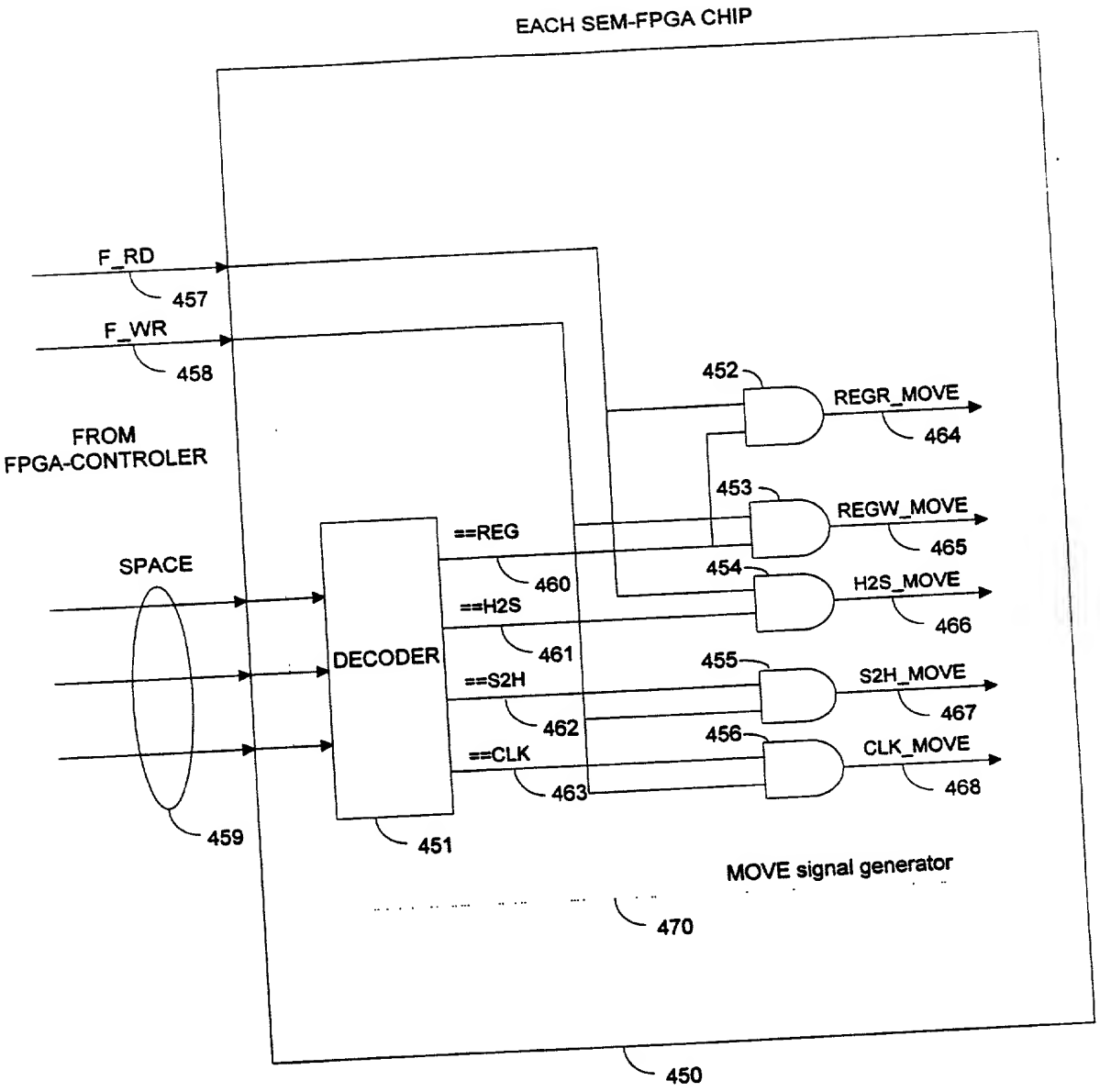

EACH SEM-FPGA CHIP
F_RD
457
F_WR
458
FROM
FPGA-CONTROLER
SPACE
459
DECODER
451
==REG
460
==H2S
461
==S2H
462
==CLK
463
452
REGR_MOVE
464
453
REGW_MOVE
465
454
H2S_MOVE
466
455
S2H_MOVE
467
456
CLK_MOVE
468
MOVE signal generator
470
450

FIG. 13

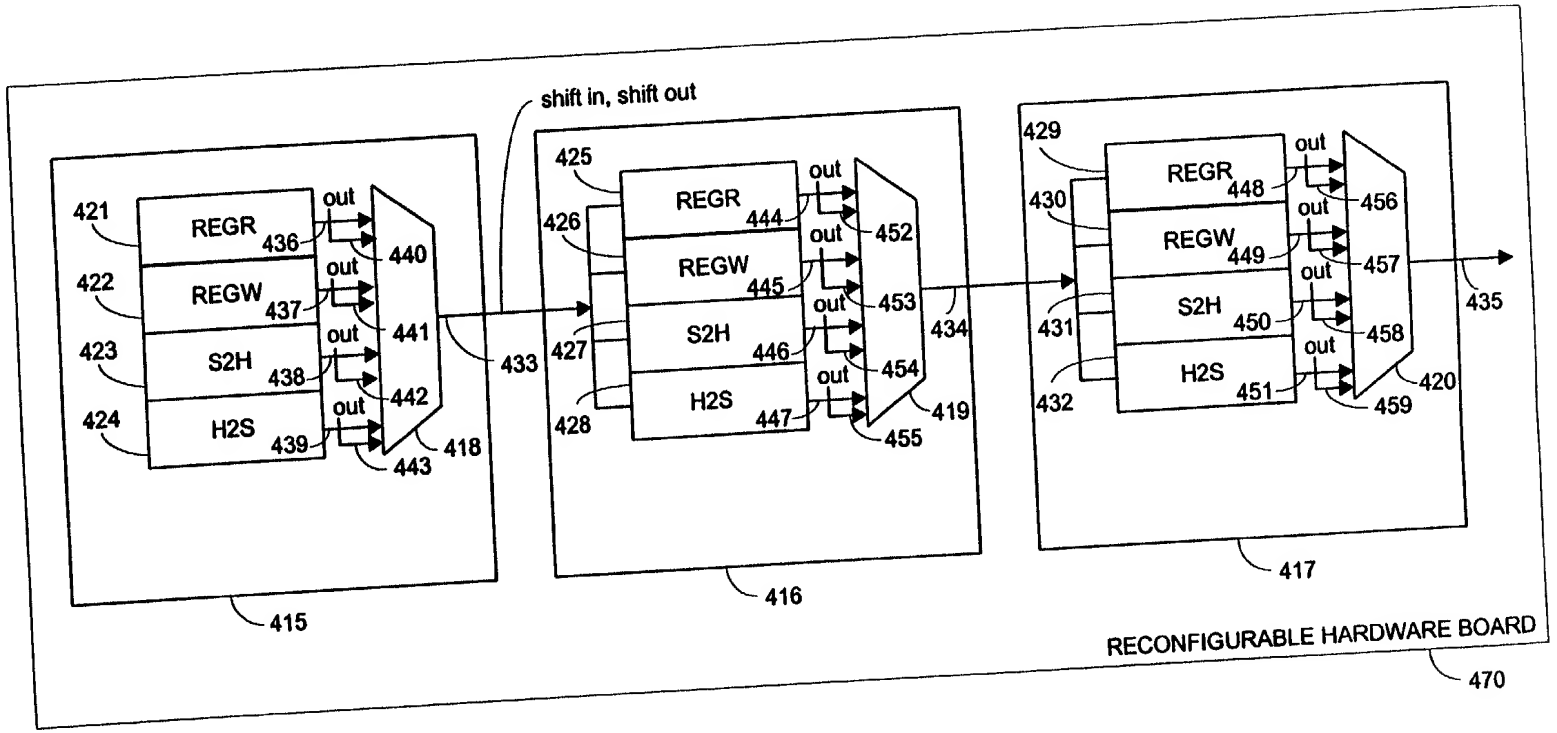

shift in, shift out
421
REGR
436
out
440
422
REGW
437
out
441
423
S2H
438
out
442
424
H2S
439
out
443
418
433
425
REGR
444
out
452
426
REGW
445
out
453
427
S2H
446
out
454
428
H2S
447
out
455
419
434
429
REGR
448
out
456
430
REGW
449
out
457
431
S2H
450
out
458
432
H2S
451
out
459
420
435
415
416
417
RECONFIGURABLE HARDWARE BOARD
470

FIG. 14

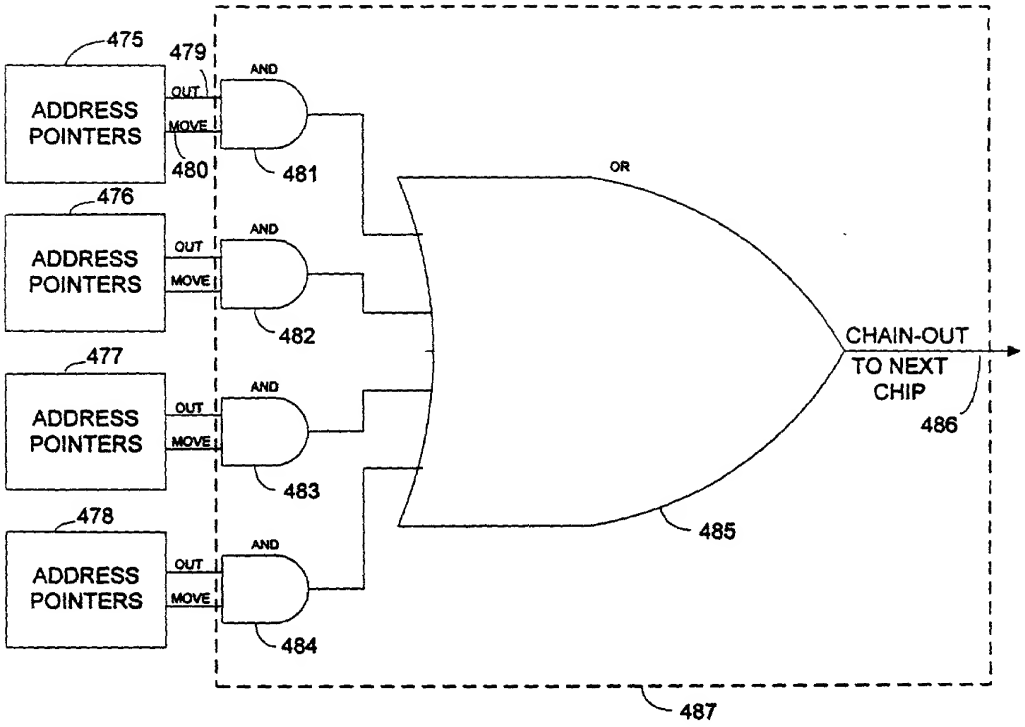

475
479
ADDRESS POINTERS
OUT
MOVE
480
AND
481
476
ADDRESS POINTERS
OUT
MOVE
AND
482
477
ADDRESS POINTERS
OUT
MOVE
AND
483
478
ADDRESS POINTERS
OUT
MOVE
AND
484
OR
CHAIN-OUT
TO NEXT CHIP
486
485
487

FIG. 15

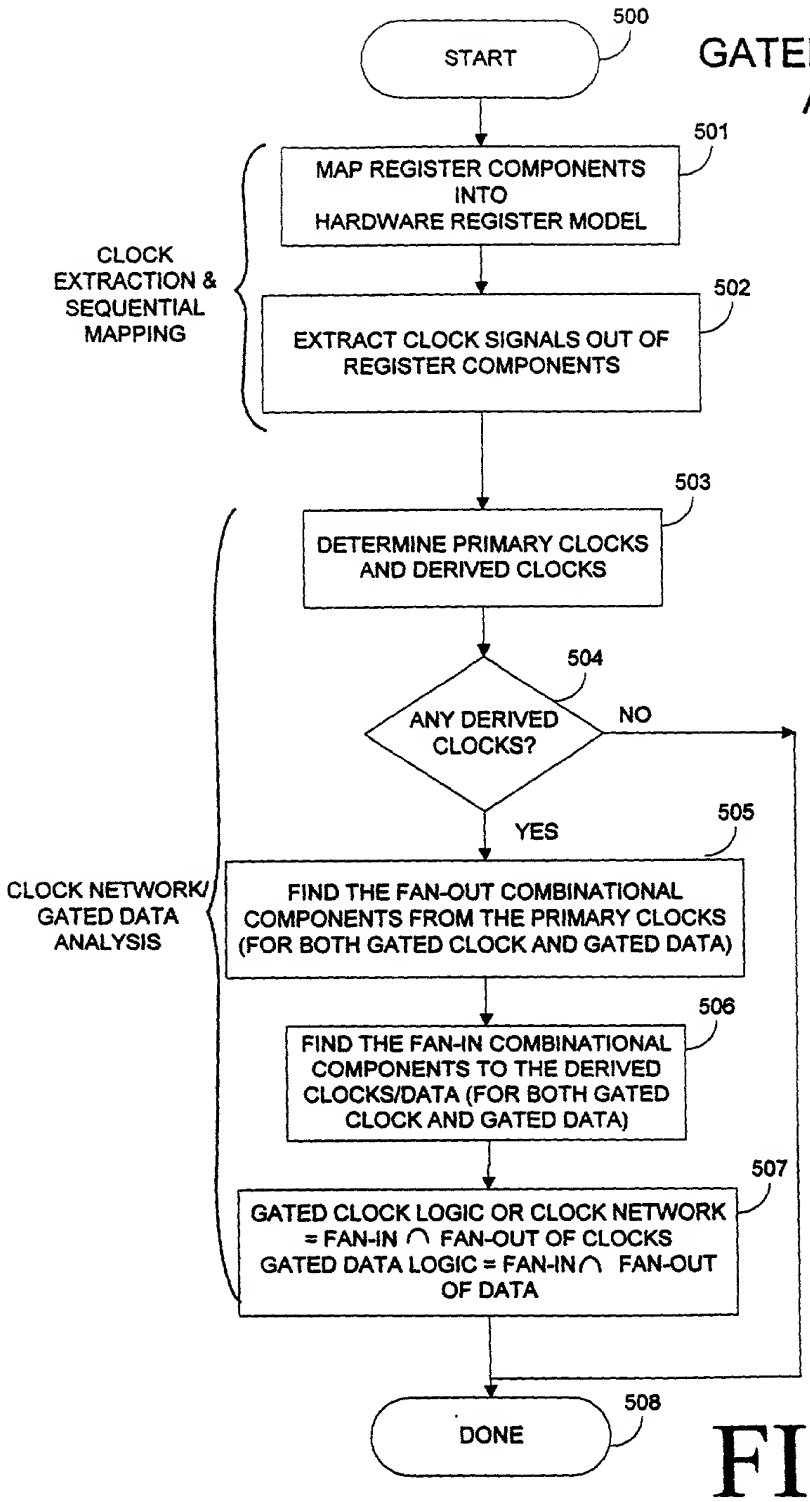

GATED DATA/CLOCK ANALYSIS
500
START
501
MAP REGISTER COMPONENTS INTO HARDWARE REGISTER MODEL
CLOCK EXTRACTION & SEQUENTIAL MAPPING
502
EXTRACT CLOCK SIGNALS OUT OF REGISTER COMPONENTS
503
DETERMINE PRIMARY CLOCKS AND DERIVED CLOCKS
504
ANY DERIVED CLOCKS?
NO
YES
505
CLOCK NETWORK/ GATED DATA ANALYSIS
FIND THE FAN-OUT COMBINATIONAL COMPONENTS FROM THE PRIMARY CLOCKS (FOR BOTH GATED CLOCK AND GATED DATA)
506
FIND THE FAN-IN COMBINATIONAL COMPONENTS TO THE DERIVED CLOCKS/DATA (FOR BOTH GATED CLOCK AND GATED DATA)
507
GATED CLOCK LOGIC OR CLOCK NETWORK = FAN-IN ∩ FAN-OUT OF CLOCKS
GATED DATA LOGIC = FAN-IN∩ FAN-OUT OF DATA
508
DONE

FIG. 16

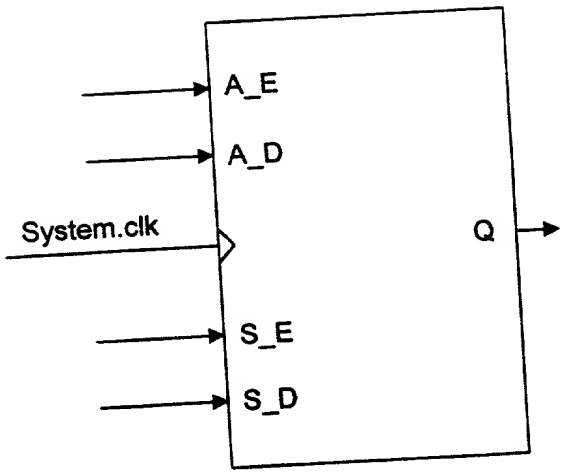
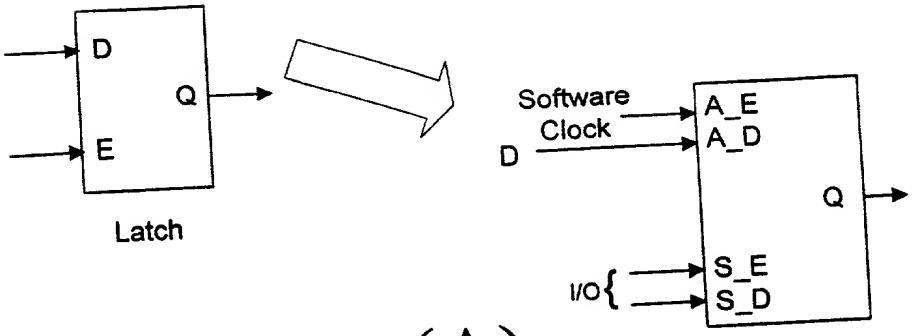
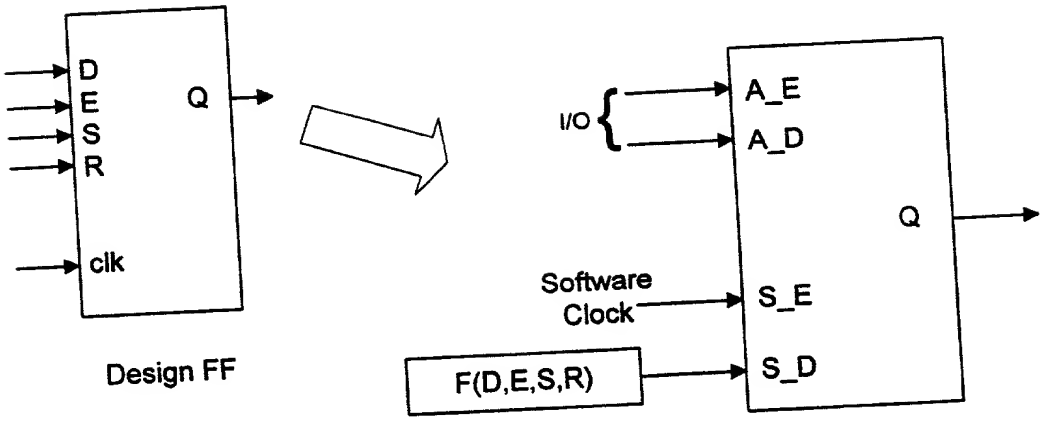


FIG. 17


D
Q
E
Latch
Software
Clock
D
A_E
A_D
Q
I/O
S_E
S_D

(A)


D
E
S
R
Q
clk
Design FF
I/O
A_E
A_D
Q
Software
Clock
S_E
F(D,E,S,R)
S_D

(B)

FIG. 18

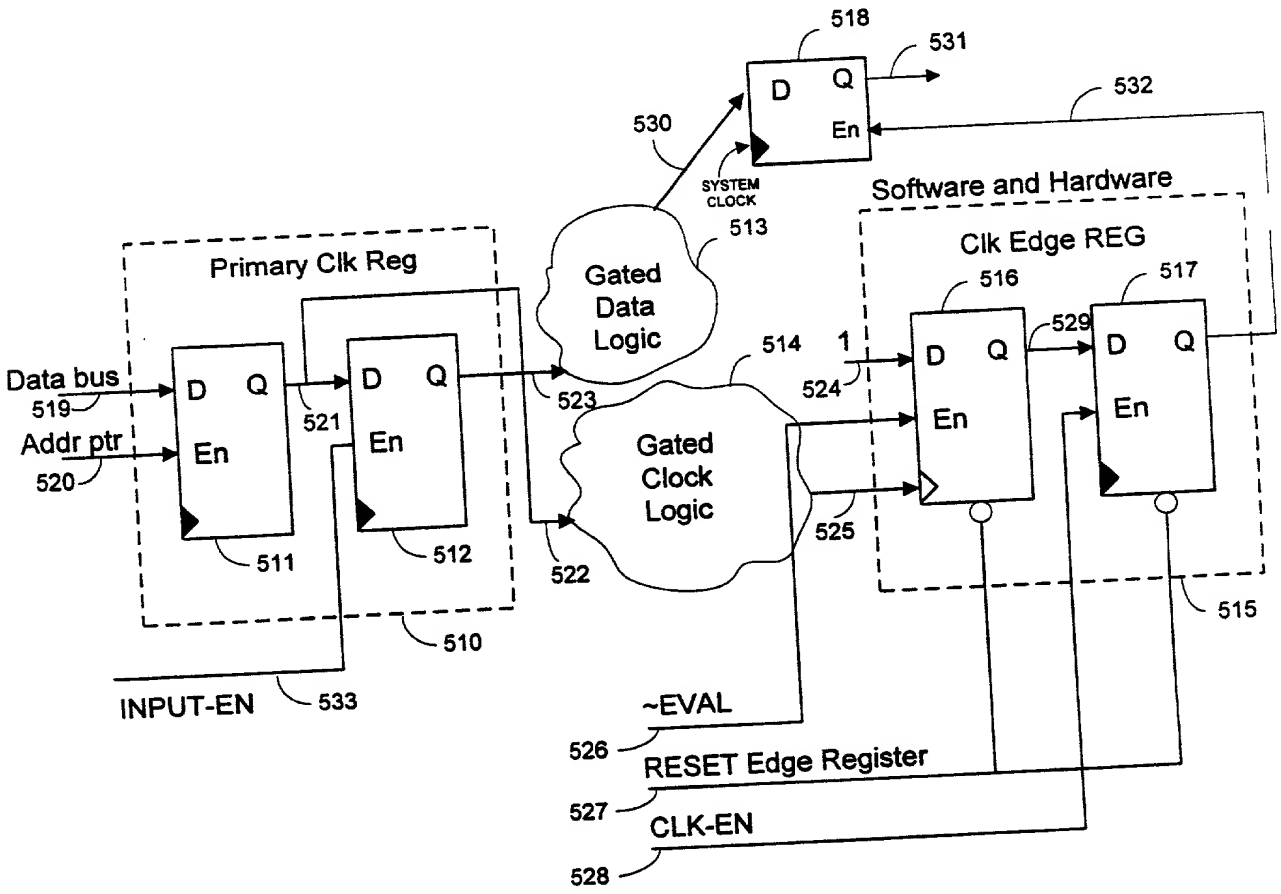

518
531
532
530
D
Q
En
SYSTEM CLOCK
Software and Hardware
513
Clk Edge REG
Primary Clk Reg
Gated Data Logic
516
517
514
1
529
Data bus
519
Addr ptr
520
D
Q
En
521
523
524
522
Gated Clock Logic
525
511
512
510
515
INPUT-EN
533
~EVAL
526
RESET Edge Register
527
CLK-EN
528

FIG. 19

DURING
EVALUATION

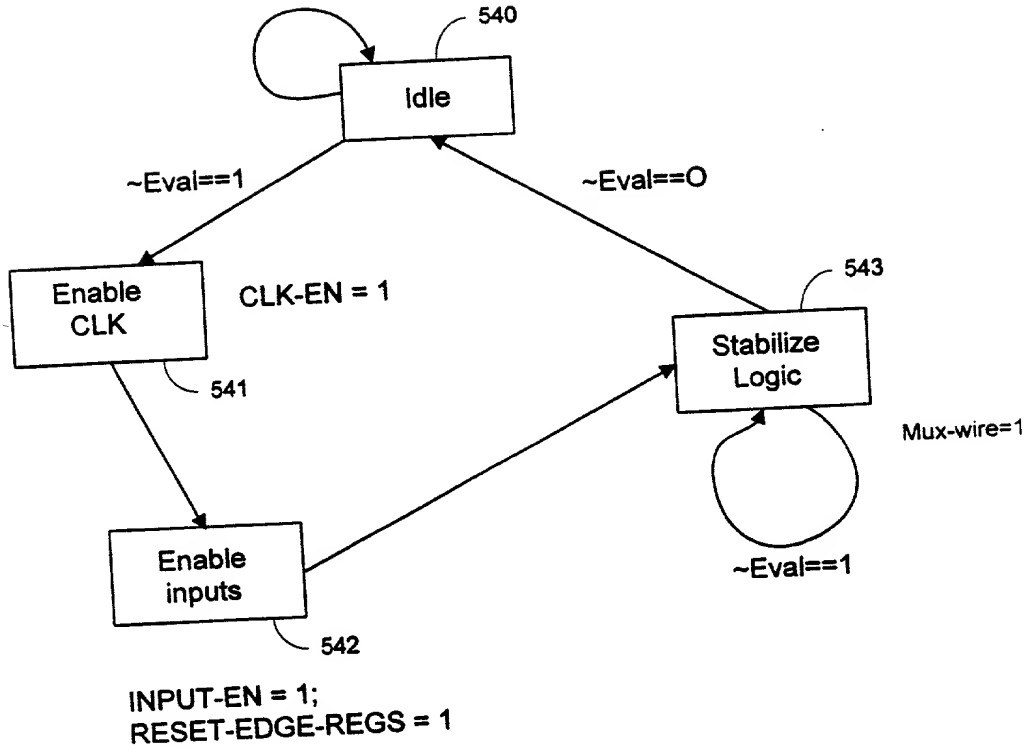


FIG. 20

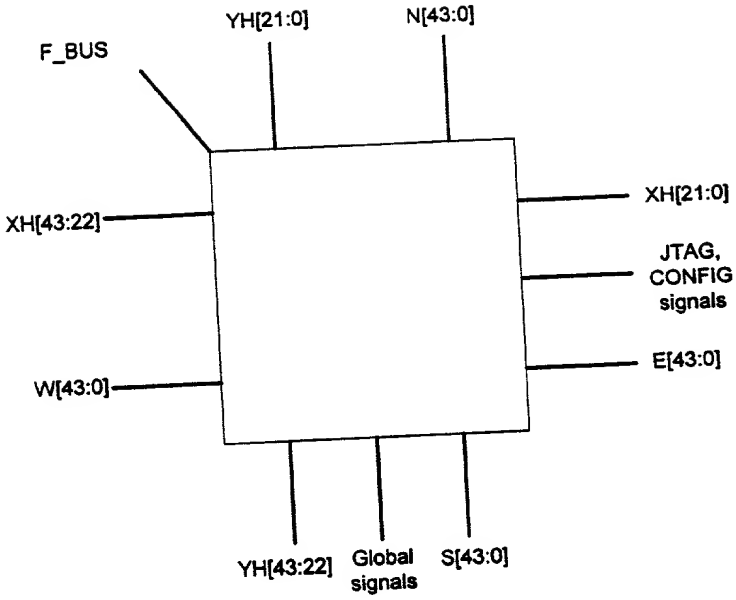

F_BUS
YH[21:0]
N[43:0]
XH[43:22]
XH[21:0]
JTAG, CONFIG signals
E[43:0]
W[43:0]
YH[43:22]
Global signals
S[43:0]

FIG. 21

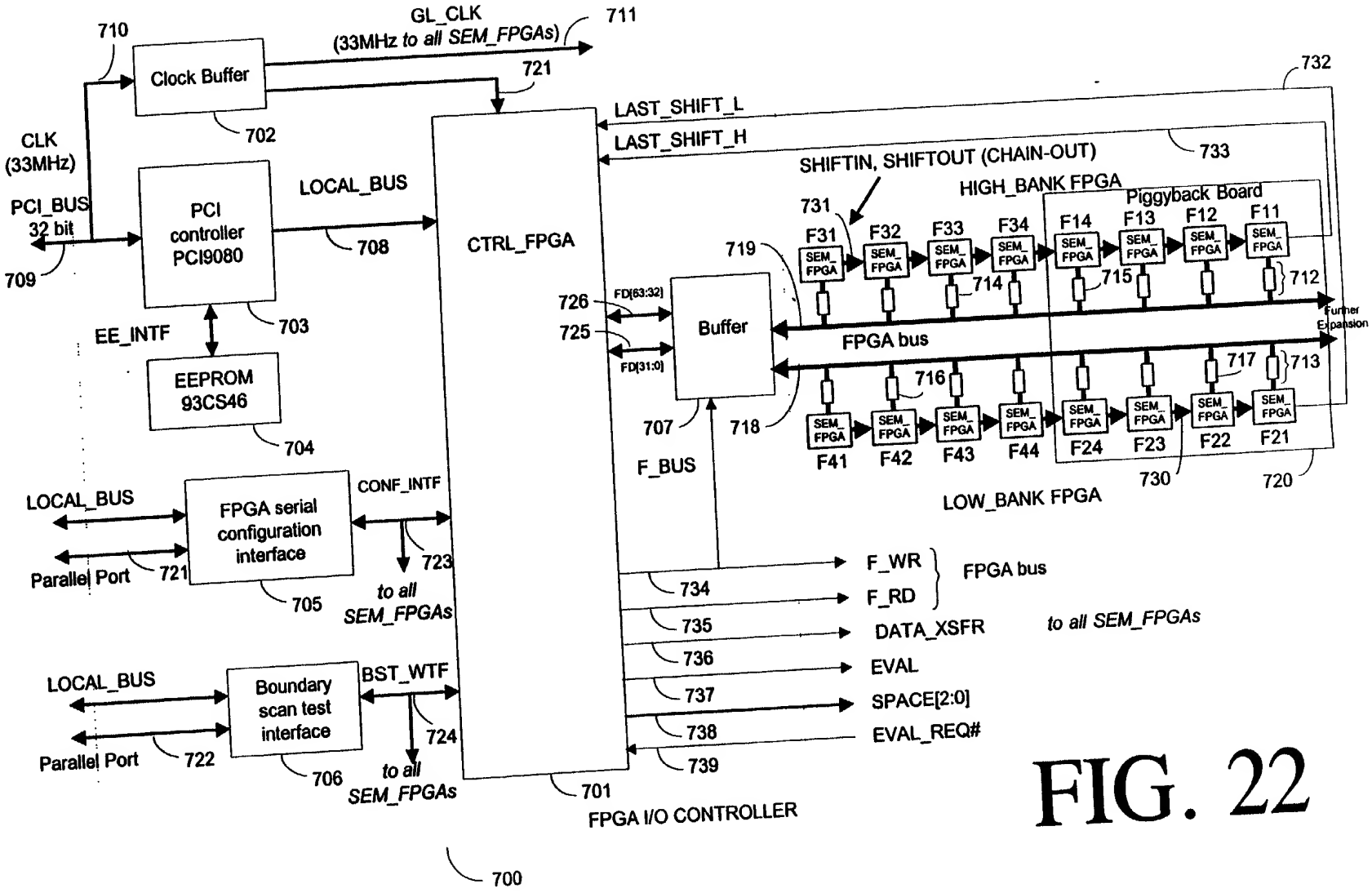

710
Clock Buffer
GL_CLK
(33MHz to all SEM_FPGAs)
711
721
732
LAST_SHIFT_L
LAST_SHIFT_H
SHIFTIN, SHIFTOUT (CHAIN-OUT)
733
HIGH_BANK FPGA
Piggyback Board
CLK
(33MHz)
PCI_BUS
32 bit
709
PCI
controller
PCI9080
LOCAL_BUS
708
CTRL_FPGA
731
719
F31
F32
F33
F34
F14
F13
F12
F11
SEM_
FPGA
714
715
712
726
FD[63:32]
725
FD[31:0]
Buffer
FPGA bus
Further
Expansion
EE_INTF
703
EEPROM
93CS46
704
716
717
713
707
718
F_BUS
F41
F42
F43
F44
F24
F23
F22
F21
730
720
LOW_BANK FPGA
LOCAL_BUS
Parallel Port
721
FPGA serial
configuration
interface
CONF_INTF
723
to all
SEM_FPGAs
705
F_WR
F_RD
FPGA bus
734
735
DATA_XSFR
to all SEM_FPGAs
736
EVAL
737
SPACE[2:0]
738
EVAL_REQ#
739
LOCAL_BUS
Parallel Port
722
Boundary
scan test
interface
BST_WTF
724
706
to all
SEM_FPGAs
701
FPGA I/O CONTROLLER
700

FIG. 22

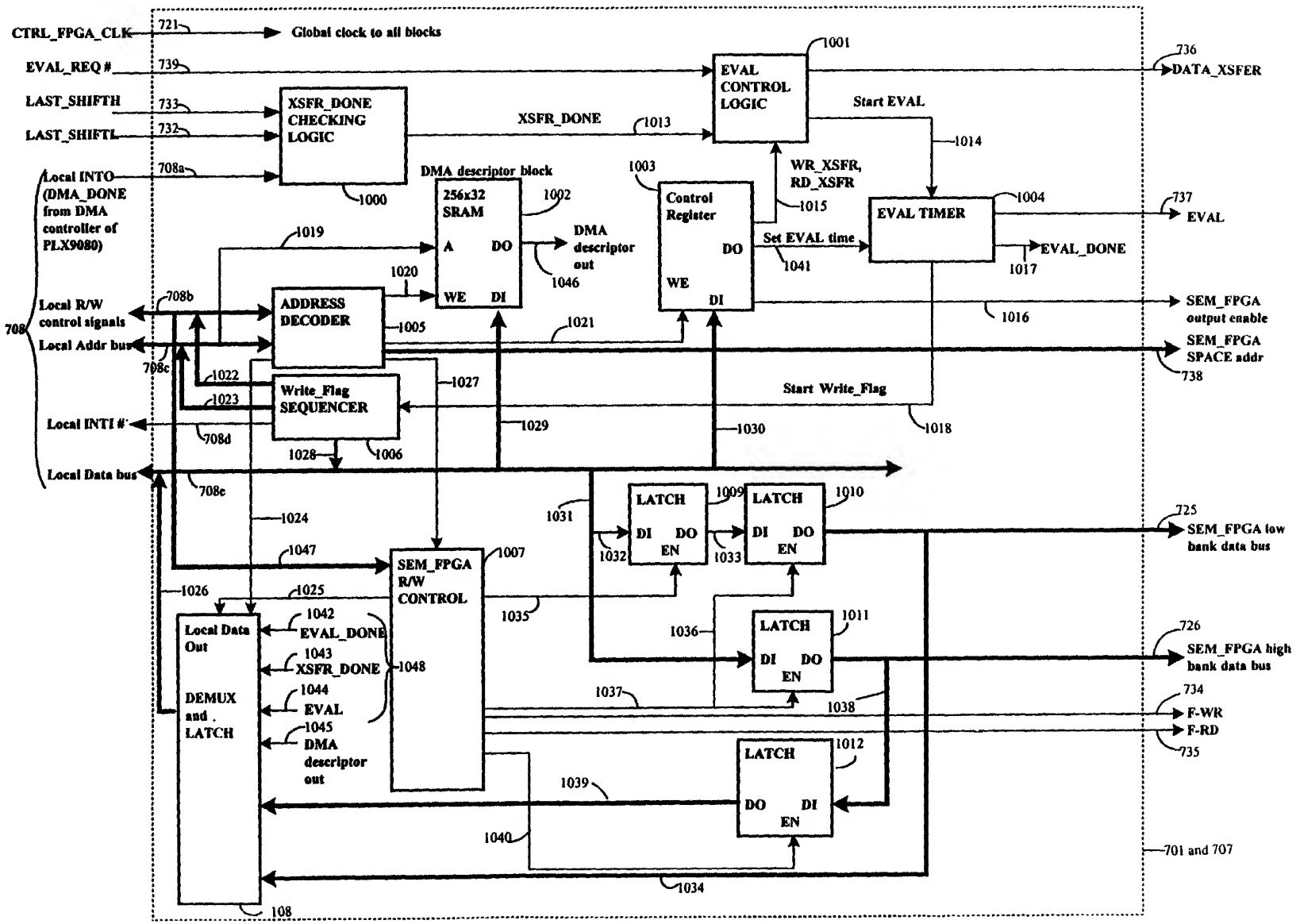


FIG. 23

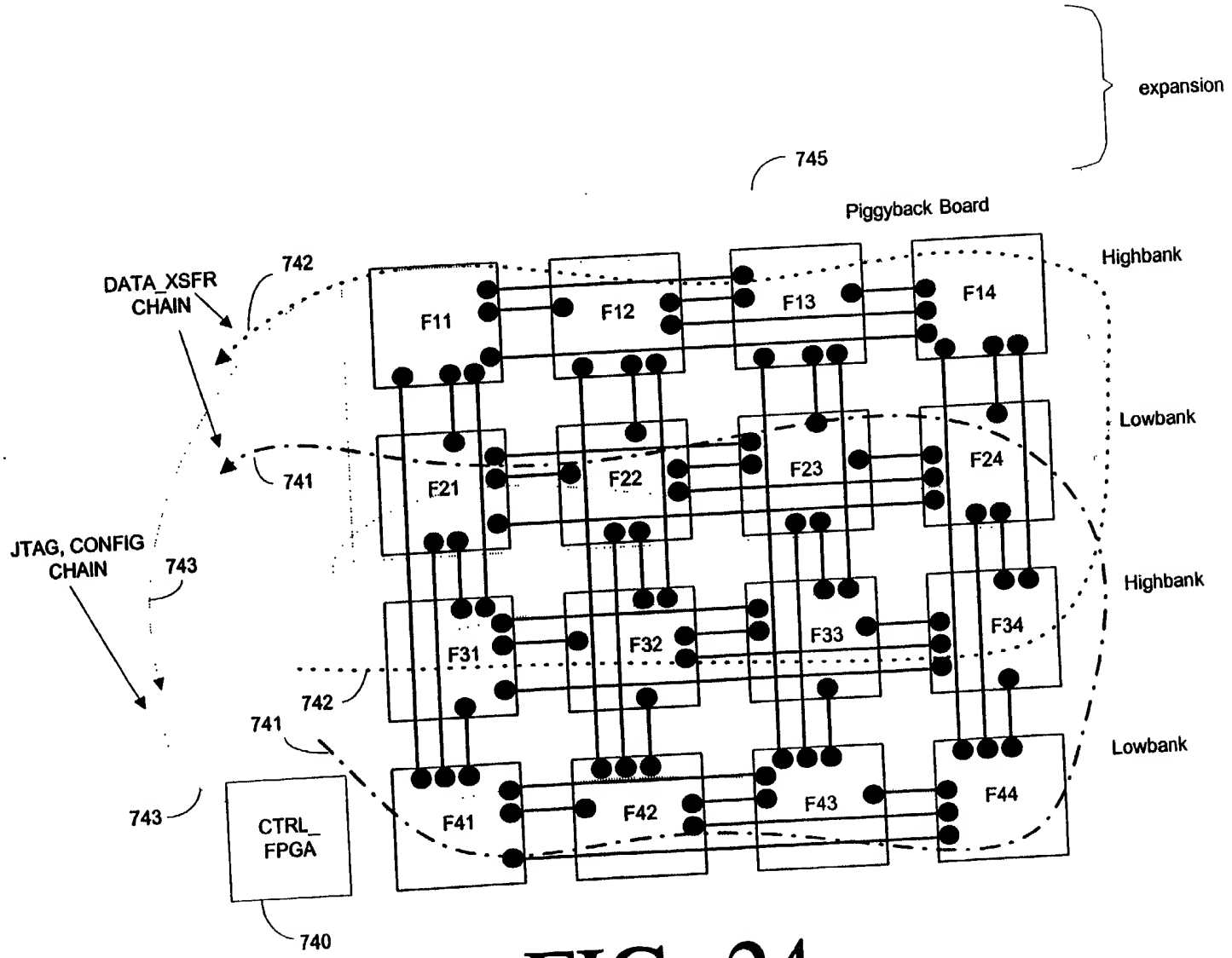

expansion
745
Piggyback Board
Highbank
Lowbank
Highbank
Lowbank
742
DATA_XSFR CHAIN
741
JTAG, CONFIG CHAIN
743
742
741
743
CTRL_ FPGA
740
F11
F12
F13
F14
F21
F22
F23
F24
F31
F32
F33
F34
F41
F42
F43
F44

FIG. 24

HARDWARE START-UP

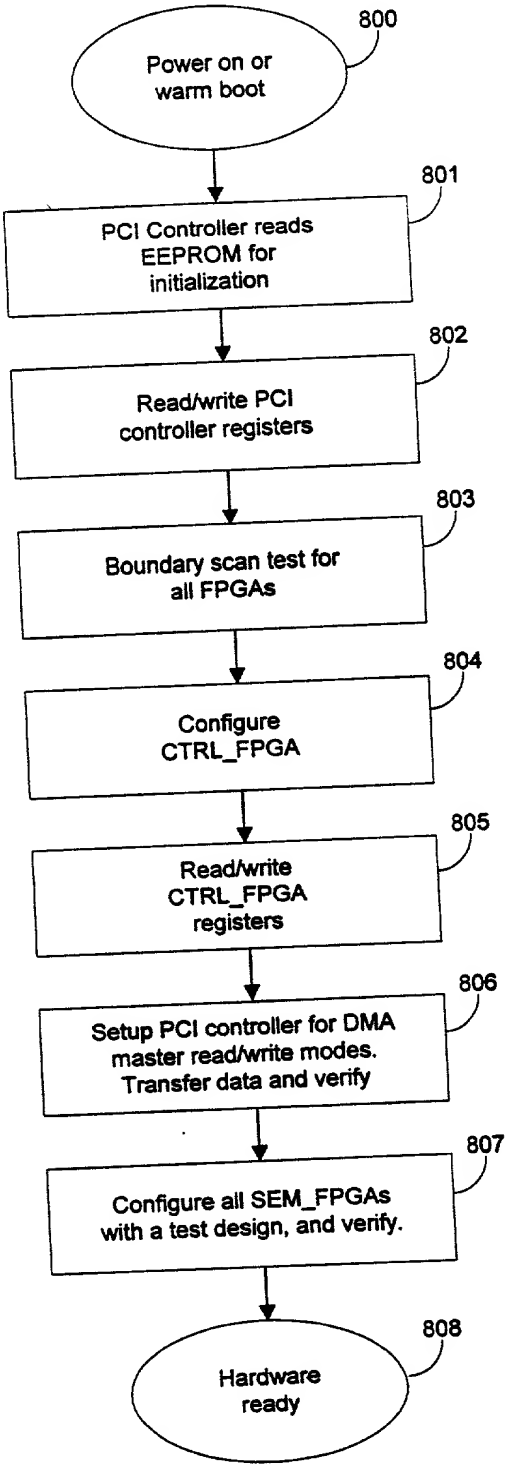


FIG. 25

```
module register (clock, reset, d, q);
input clock, d, reset;
output q;
reg q;

always@(posedge clock or negedge reset)
    if(~reset)
        q = 0;
    else
        q = d;

endmodule

module example;
    wire d1, d2, d3;
    wire q1, q2, q3;

    reg sigin;
    wire sigout;
    reg clk, reset;

    register reg1 (clk, reset, d1, q1);
    register reg2 (clk, reset, d2, q2);
    register reg3 (clk, reset, d3, q3);

    assign d1 = sigin ^ q3;
    assign d2 = q1 ^ q3;
    assign d3 = q2 ^ q3;
    assign sigout = q3;

    // a clock generator
    always
    begin
        clk = 0;
        #5;
        clk = 1;
        #5;
    end

    // a signal generator
    always
    begin
        #10;
        sigin = $random;
    end

    // initialization
    initial
    begin
        reset = 0;
        sigin = 0;
        #1;
        reset =1;
        #5;
        $monitor($time, " %b, %b", sigin, sigout);
        #1000 $finish;
    end
    end module
```

FIG. 26

CIRCUIT DIAGRAM

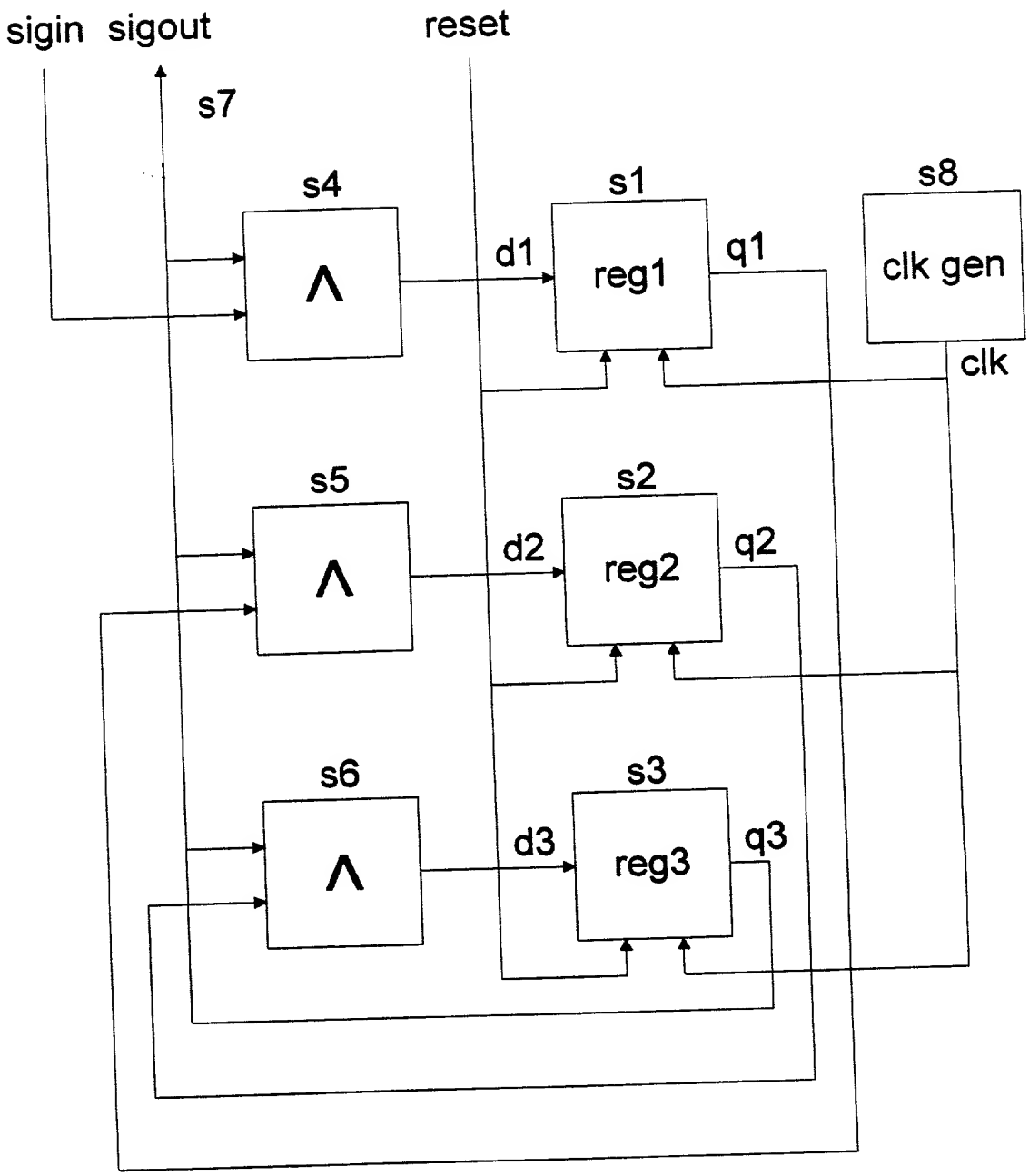


FIG. 27

```
module register (clock, reset, d, q);
input clock, d, reset;
output q;
reg q;

always@(postedge clock or negedge reset)
   if(~reset)
      q = 0
   else
      q = d;

endmodule
```

Register Definition — 900

```
module example;
   wire d1, d2, d3;
   ware q1, q2, q3;
```

wire interconnection info — 907

```
   reg sigin;
   wire sigout;
   reg clk, reset;
```

Test-bench input – 908
Test-bench output – 909

```
S1 register reg 1 (clk, reset, d1, q1);
S2 register reg 2 (clk, reset, d2, q2);
S3 register reg 3 (clk, reset, d3, q3);
```

Register component — 901

```
S4 assign d1 = sigin ^ q3;
S5 assign d2 = q1 ^ 3;
S6 assign d3 = q2 ^ q3;
S7 assign signout = q3;
```

Combinational component — 902

```
S8  // a clock generator
    always
    begin
       clk = 0;
       #5;
       clk = 1;
       #5;
    end
```

Clock component — 903

```
S9  // a signal generator
    always
    begin
       #10;
       sigin = $random;
    end
```

Test-bench component (Driver) — 904

```
    // initialization
    initial
    begin
S10    reset = 0;
       sigin = 0;
       #1;
S11    reset = 1;
       #5;
S12    $monitor($time, "%b, %b", sigin, sigout);
       #1000 $finish;
    end
    end module
```

Test-bench component (initialization) — 905

Test-bench component (monitor) — 906

FIG. 28

SIGNAL NETWORK ANALYSIS

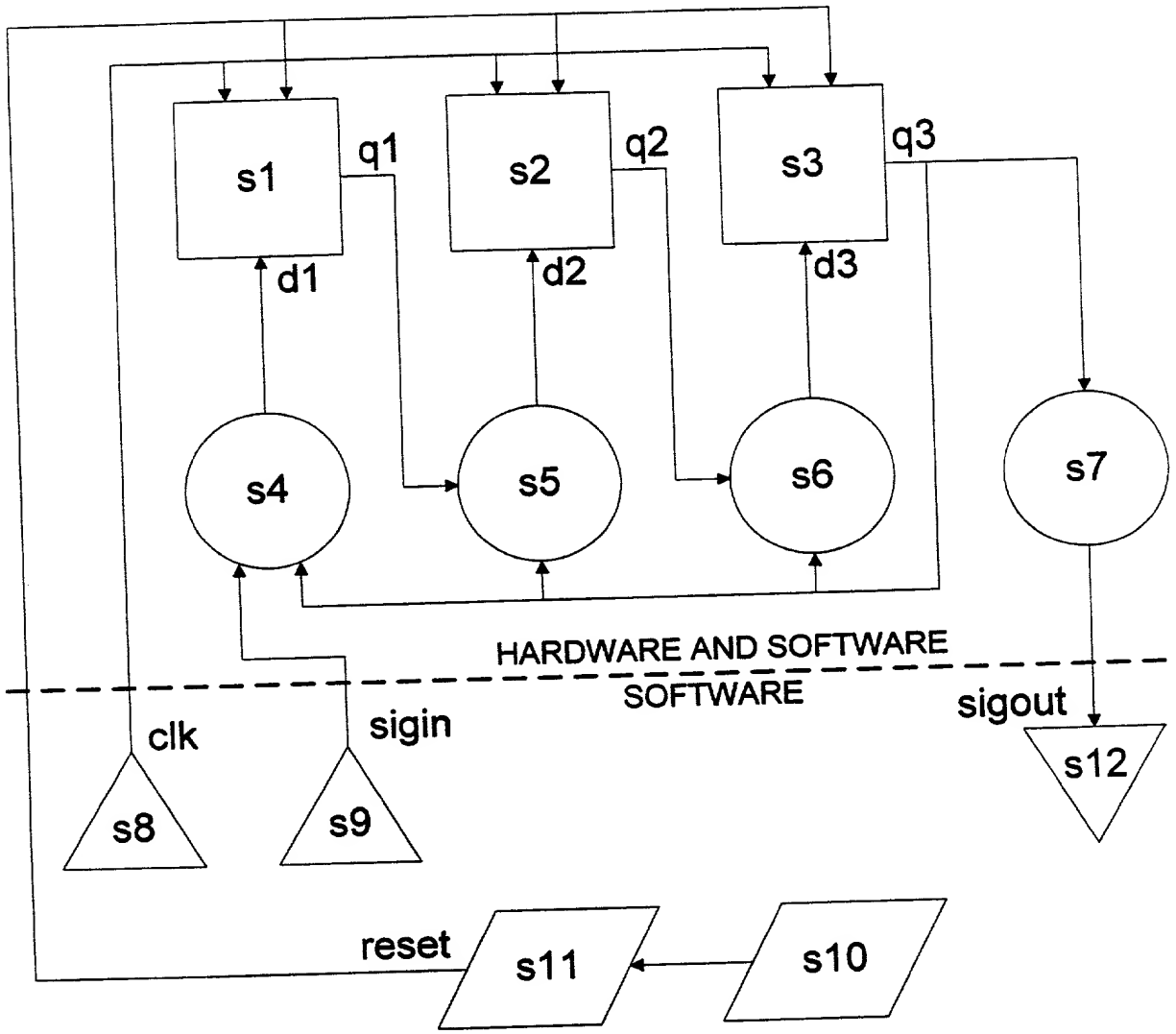


FIG. 29

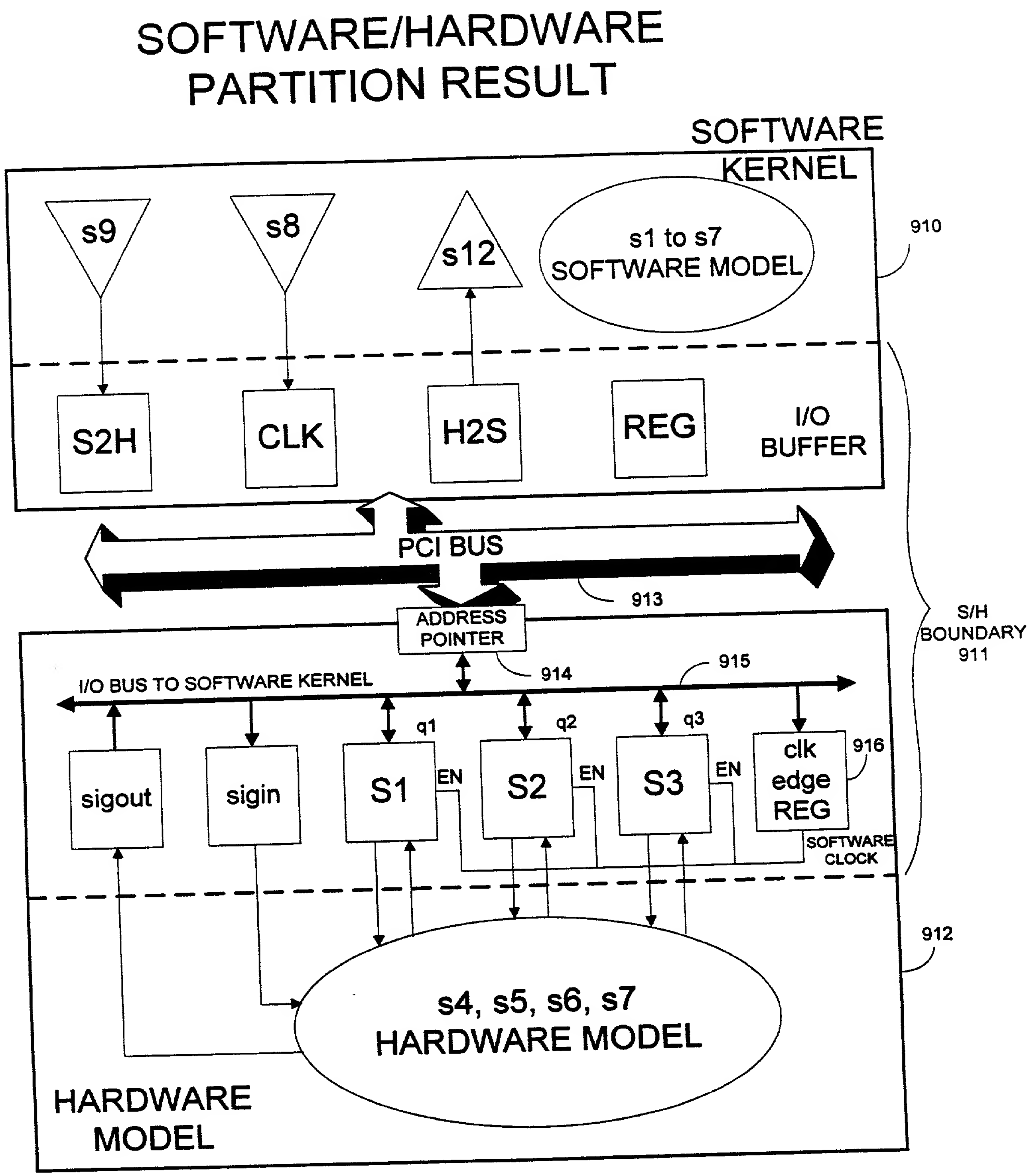

SOFTWARE/HARDWARE
PARTITION RESULT
SOFTWARE
KERNEL
s9
s8
s12
s1 to s7
SOFTWARE MODEL
910
S2H
CLK
H2S
REG
I/O
BUFFER
PCI BUS
913
S/H
BOUNDARY
911
ADDRESS
POINTER
914
915
I/O BUS TO SOFTWARE KERNEL
q1
q2
q3
sigout
sigin
S1
EN
S2
EN
S3
EN
clk
edge
REG
916
SOFTWARE
CLOCK
912
s4, s5, s6, s7
HARDWARE MODEL
HARDWARE
MODEL

FIG. 30

HARDWARE MODEL

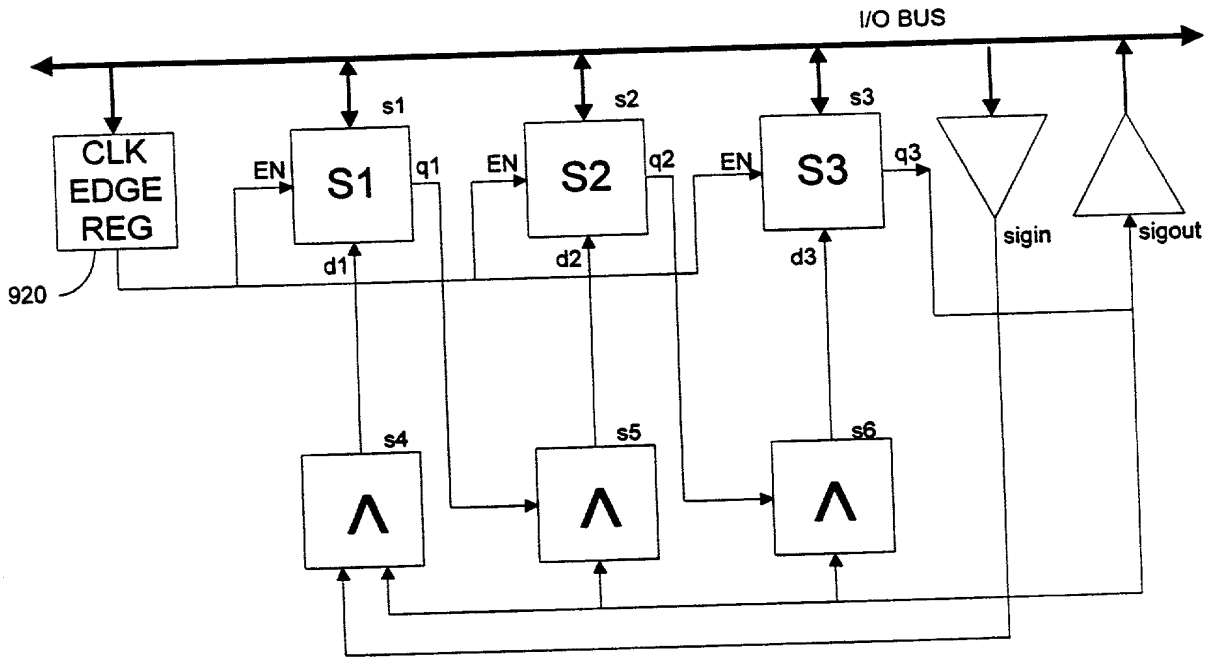
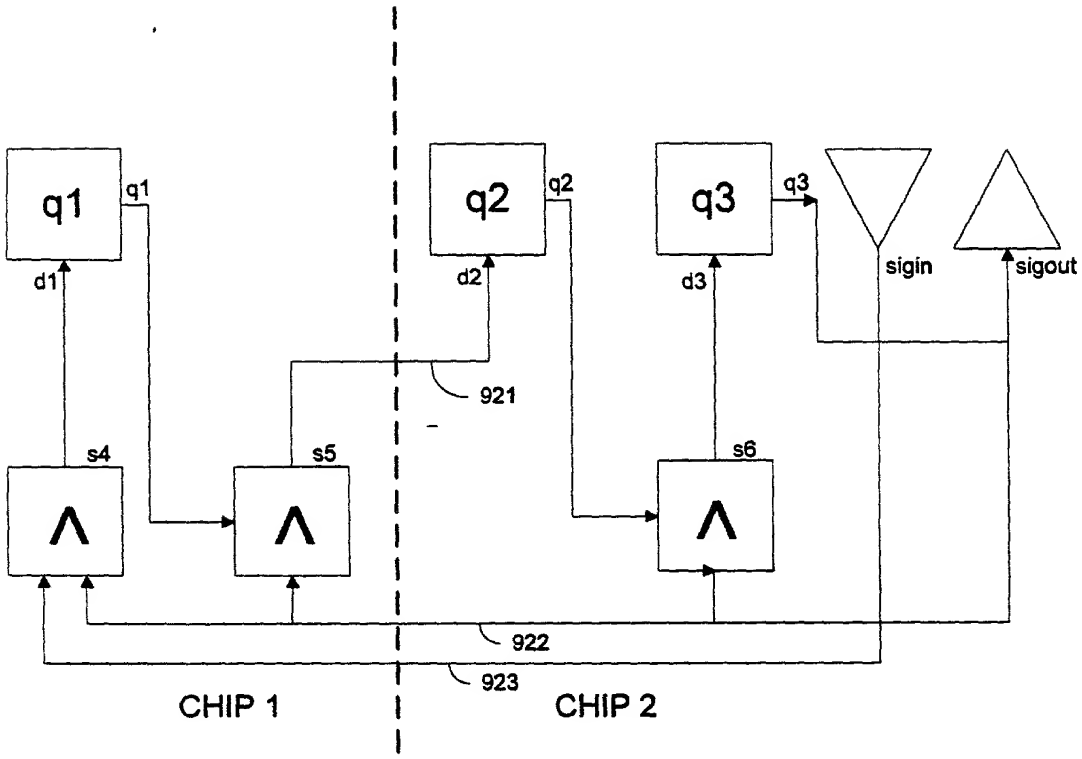


FIG. 31

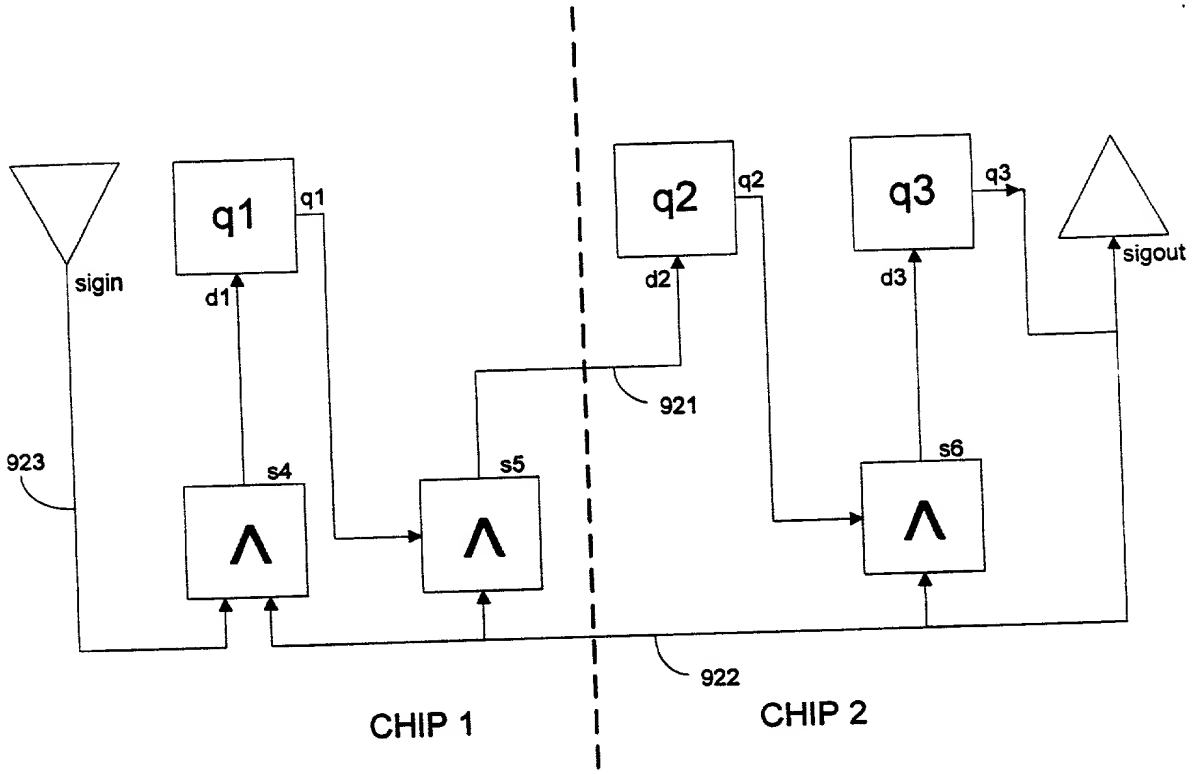
PARTITION RESULT #1



(IGNORE I/O AND CLOCK EDGE REGISTER)

FIG. 32

PARTITION RESULT #2



(IGNORE I/O AND CLOCK EDGE REGISTER)

FIG. 33

LOGIC PATCHING

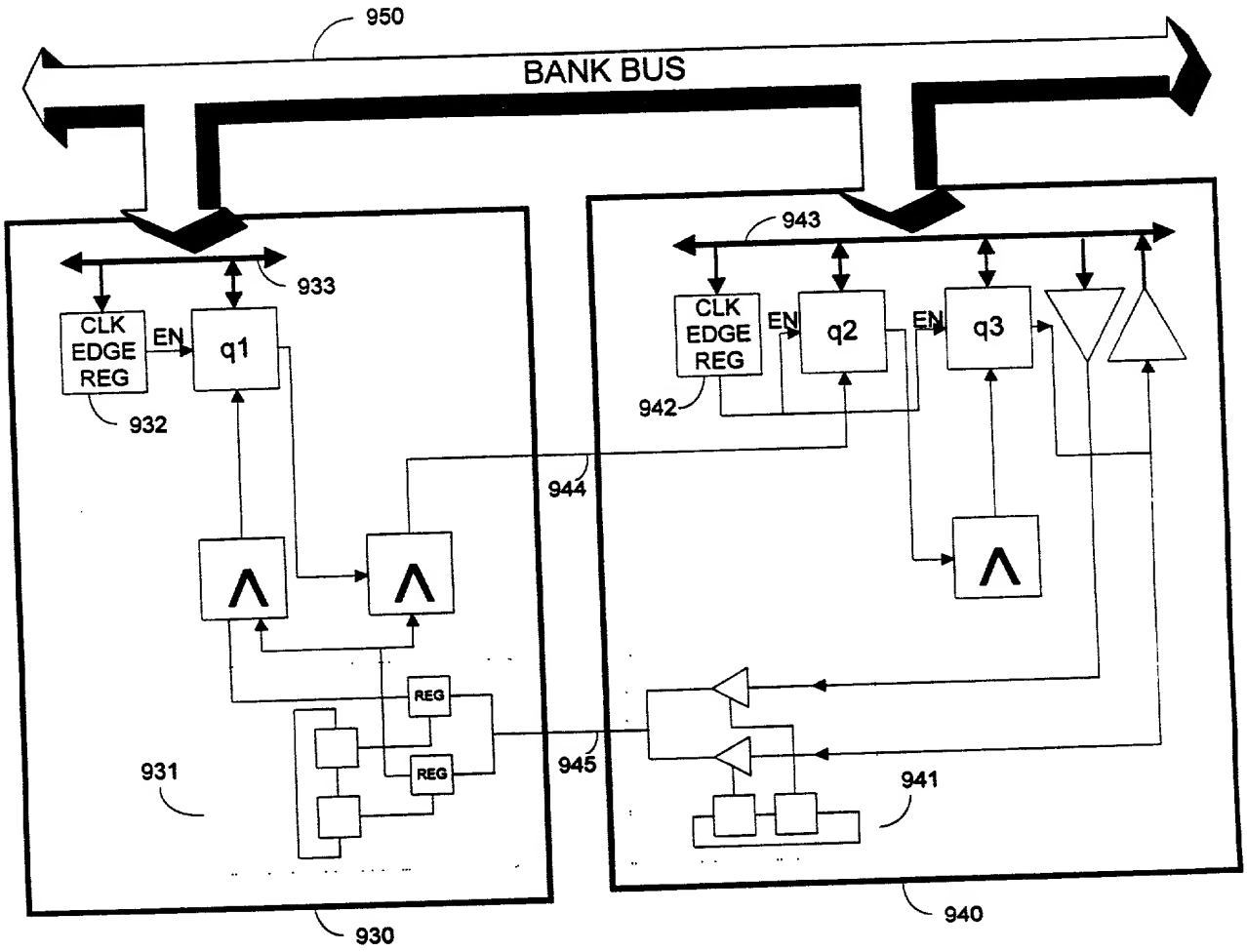
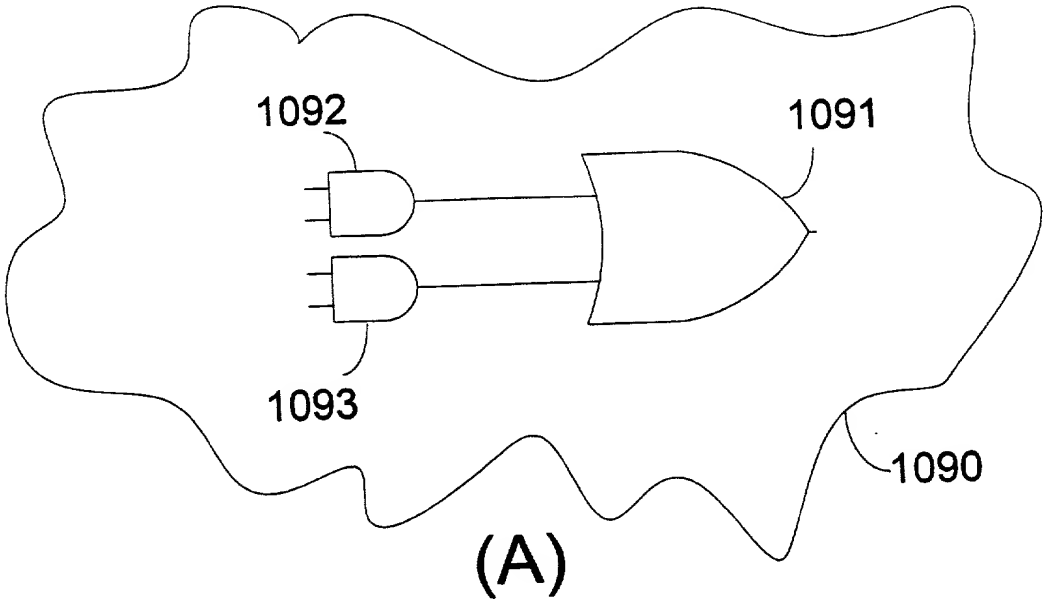
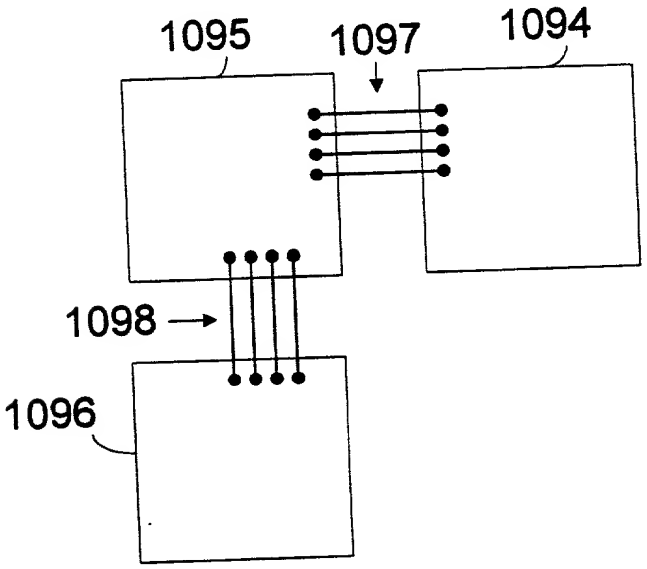


FIG. 34

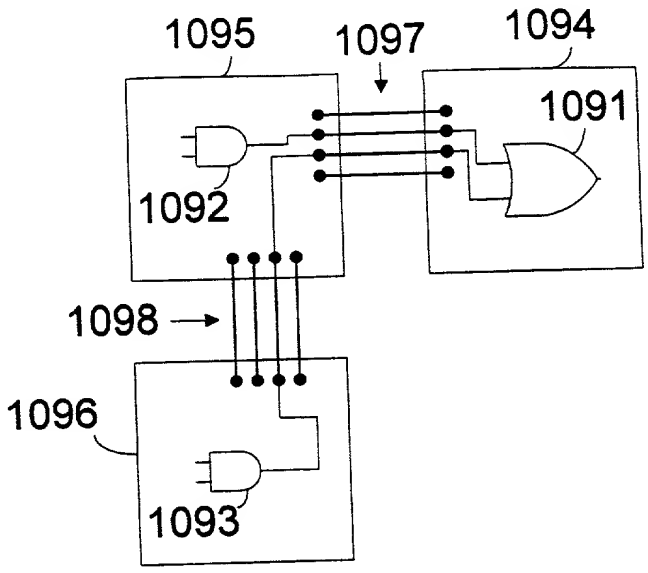


(A)

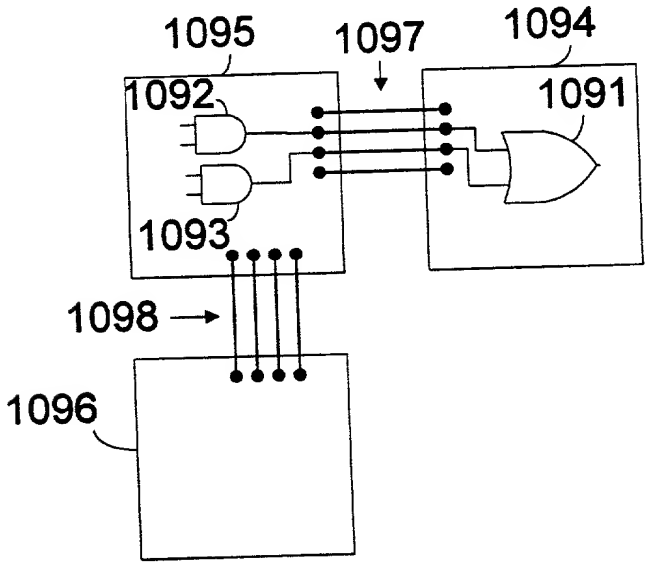


(B)

FIG. 35


1095
1097
1094
1091
1092
1098
1096
1093

(C)

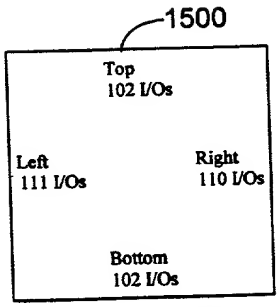

1095
1097
1094
1092
1091
1093
1098
1096

(D)

FIG. 35

I/O PIN OVERVIEW OF FPGA LOGIC DEVICE

FPGA : 10K130V, 10K250V with 599-pin PGA package



425 Interconnect I/O pins

45 Dedicated I/O pins:

GCLK, FD_ BUS[31..0], F_RD, F_WR, DATAXSFR, SHIFTIN, SHIFTOUT, SPACE[2..0], EVAL, EV_REQ_N, DEV_OE, DEV_CLRN

FIG. 36

FPGA INTERCONNECT BUSES

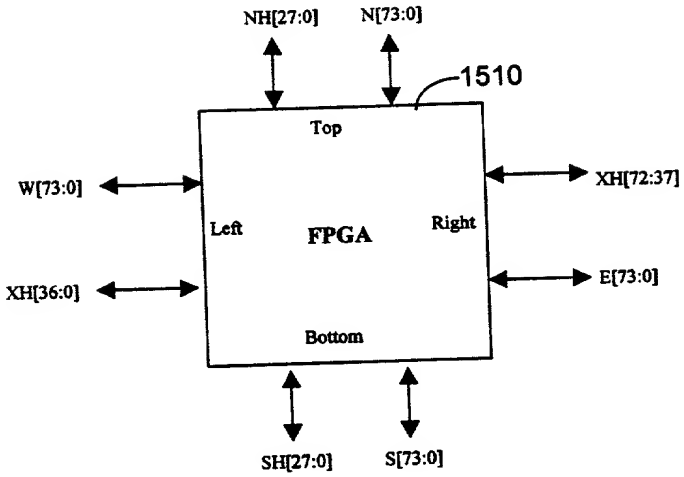


FIG. 37

BOARD CONNECTION - SIDE VIEW

DUAL-BOARD CONFIGURATION

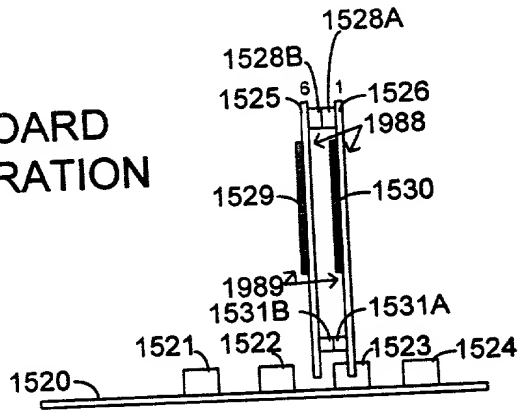


FIG. 38(A)

SIX BOARD CONFIGURATION

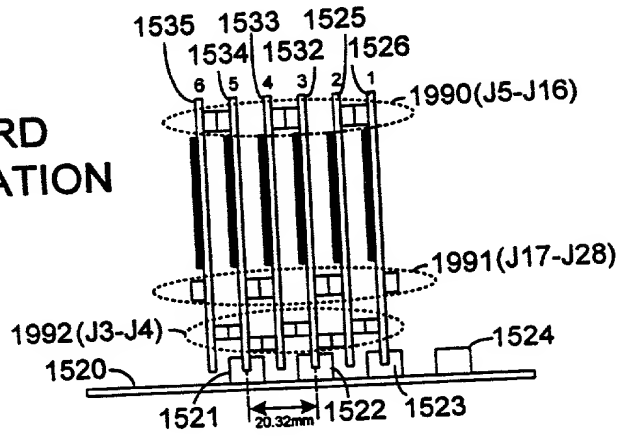


FIG. 38(B)

SIX-BOARD CONFIGURATION
DIRECT-NEIGHBOR AND ONE-HOP FPGA ARRAY – X TORUS, Y MESH

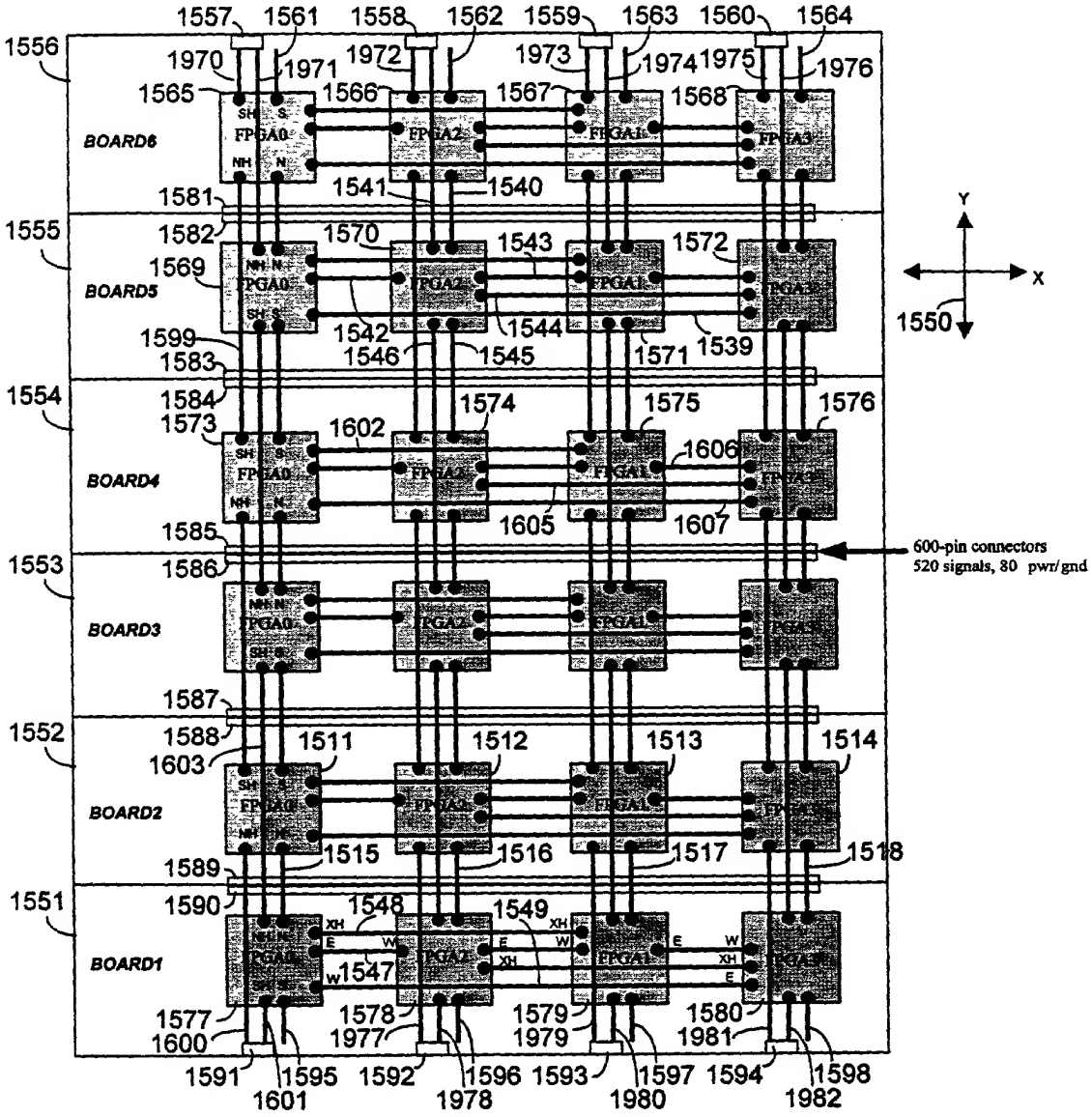


FIG. 39

FPGA ARRAY CONNECTION BETWEEN BOARDS

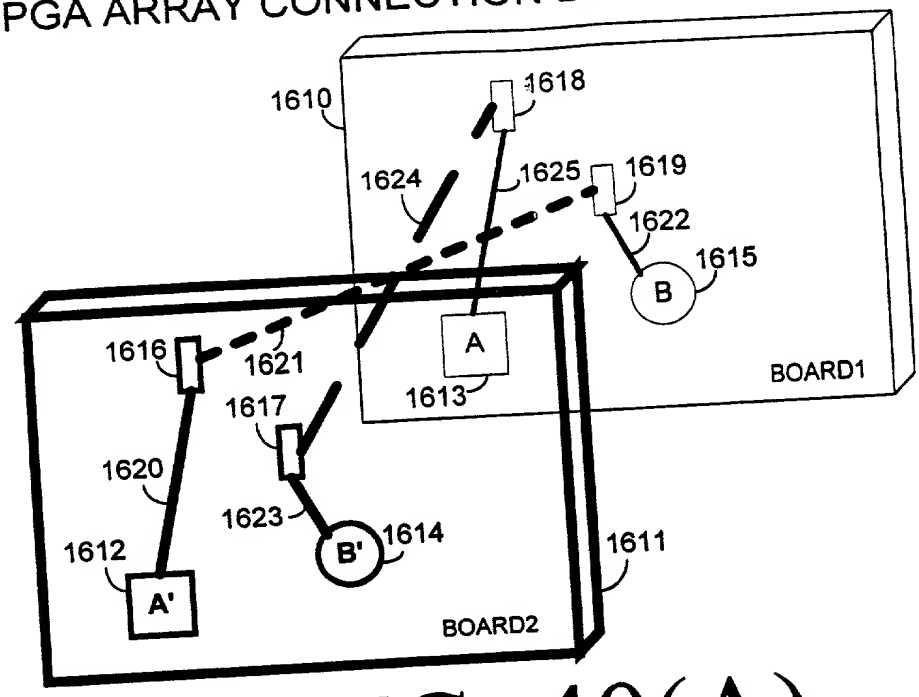


FIG. 40(A)

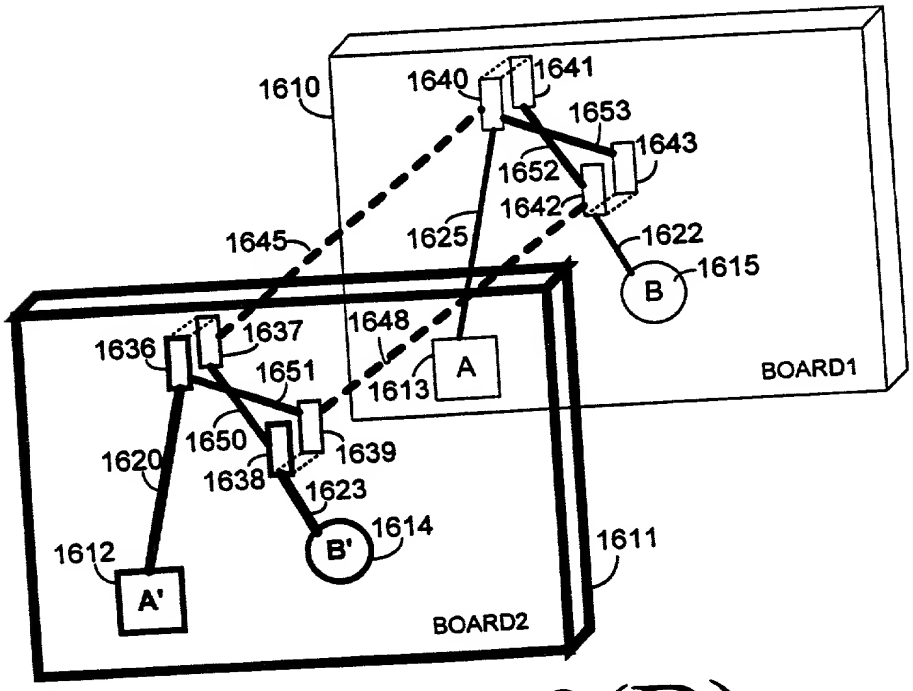


FIG. 40(B)

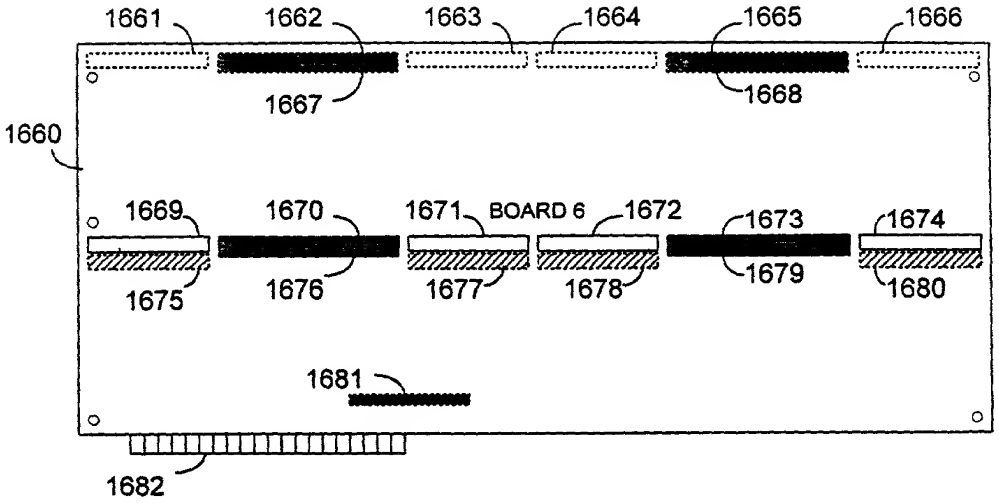


FIG. 41(A)

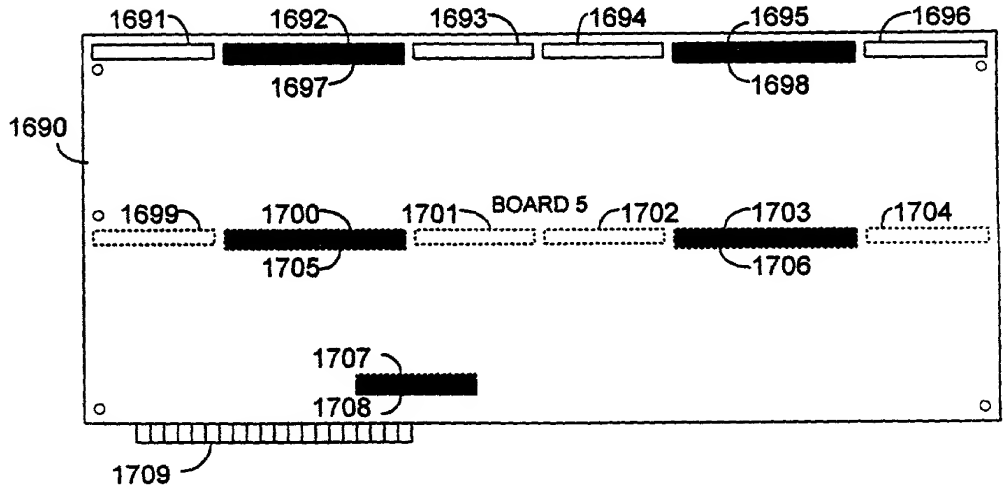


FIG. 41(B)

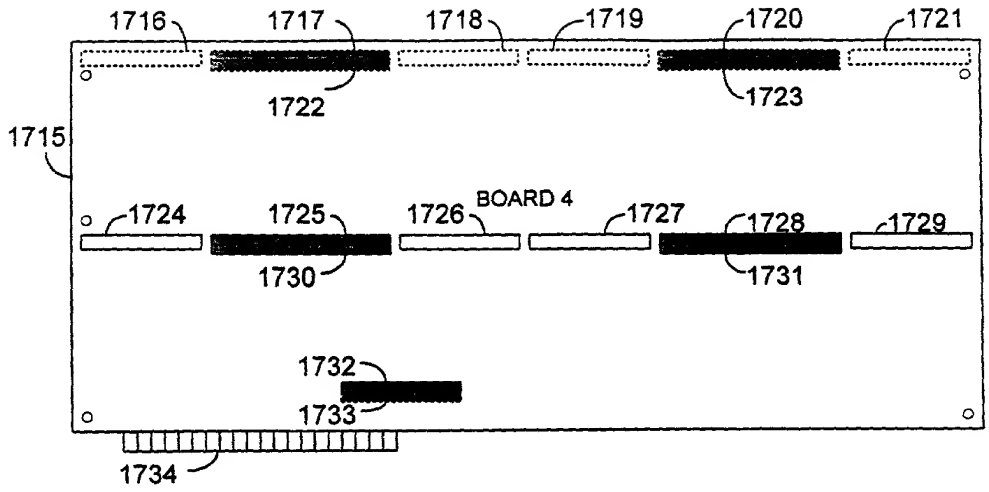


FIG. 41(C)

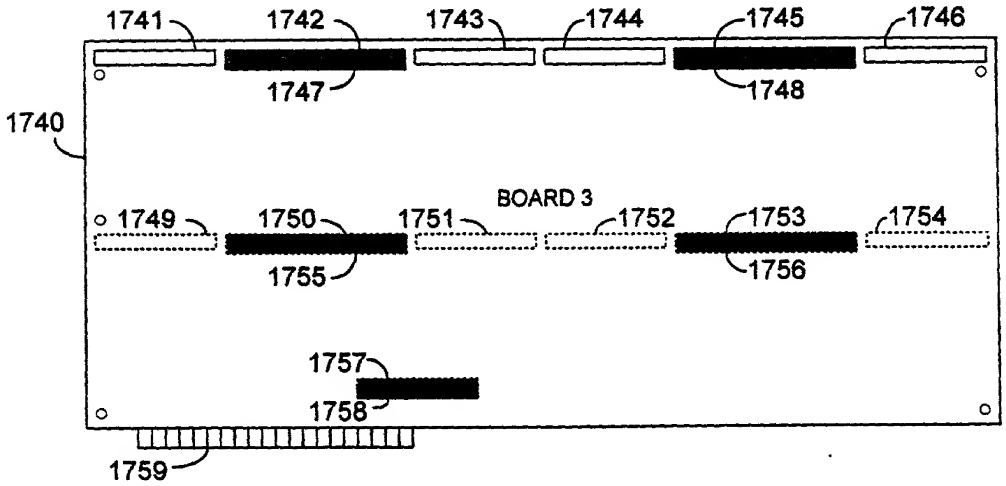


FIG. 41(D)

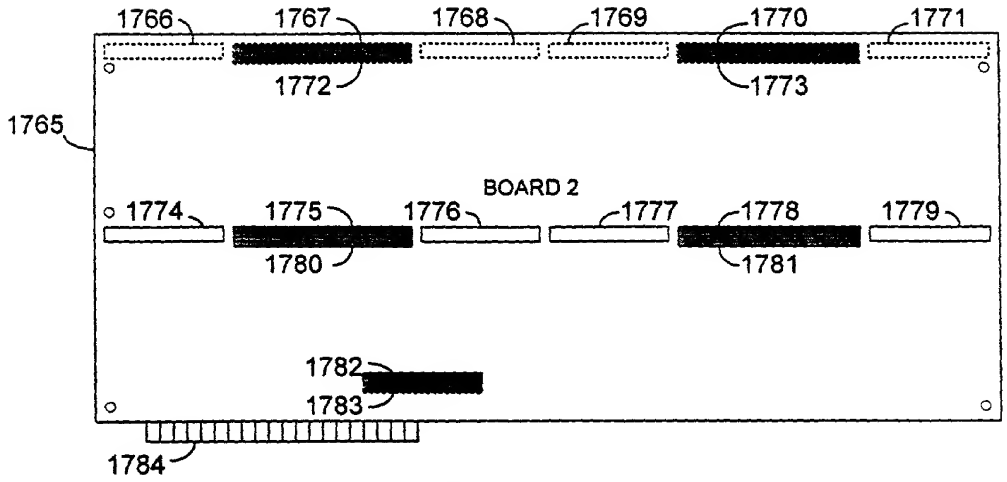


FIG. 41(E)

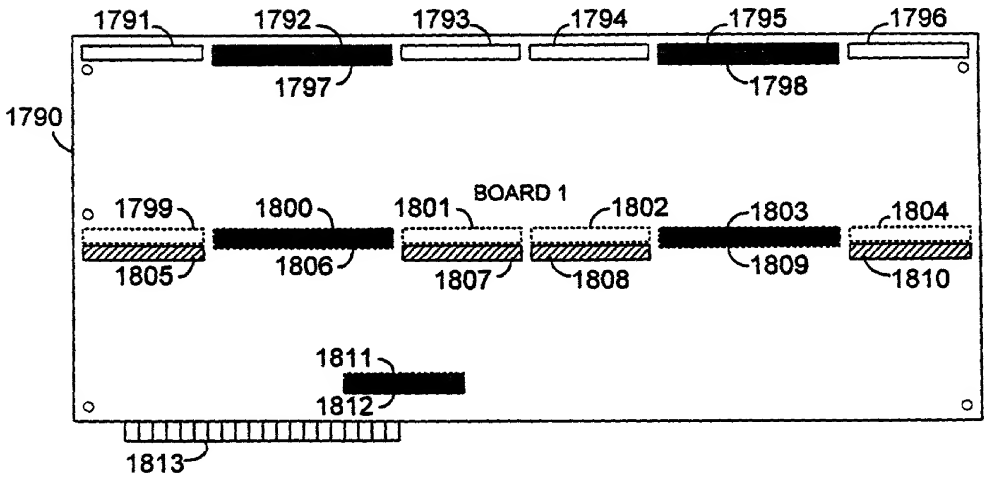


FIG. 41(F)

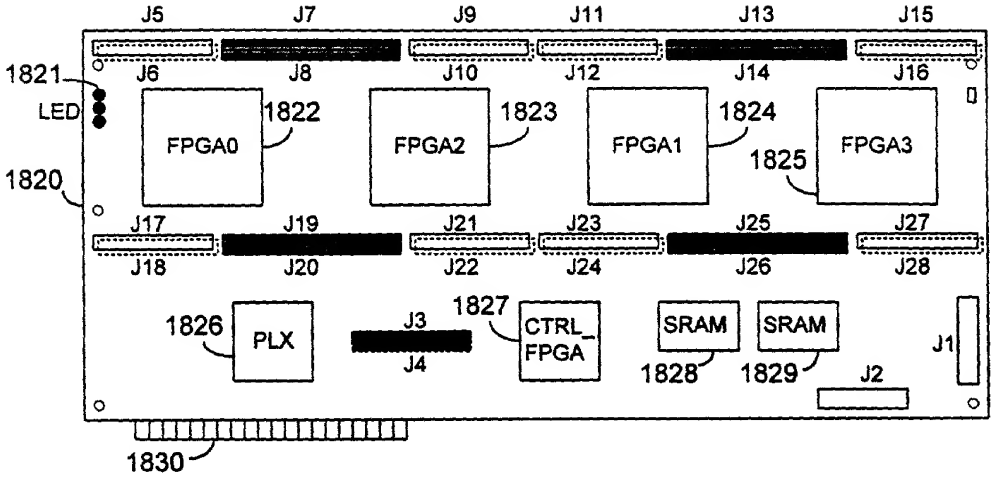

J5
J7
J9
J11
J13
J15
J6
J8
J10
J12
J14
J16
1821
LED
1820
FPGA0
1822
FPGA2
1823
FPGA1
1824
FPGA3
1825
J17
J19
J21
J23
J25
J27
J18
J20
J22
J24
J26
J28
1826
PLX
J3
J4
1827
CTRL_
FPGA
SRAM
SRAM
1828
1829
J2
J1
1830

FIG. 42

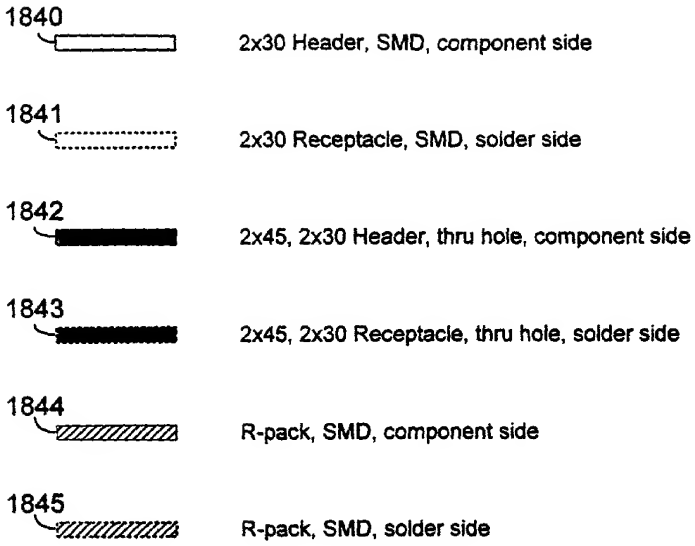

1840
2x30 Header, SMD, component side
1841
2x30 Receptacle, SMD, solder side
1842
2x45, 2x30 Header, thru hole, component side
1843
2x45, 2x30 Receptacle, thru hole, solder side
1844
R-pack, SMD, component side
1845
R-pack, SMD, solder side

FIG. 43

TWO-BOARD CONFIGURATION
DIRECT-NEIGHBOR AND ONE-HOP FPGA ARRAY – X TORUS, Y MESH

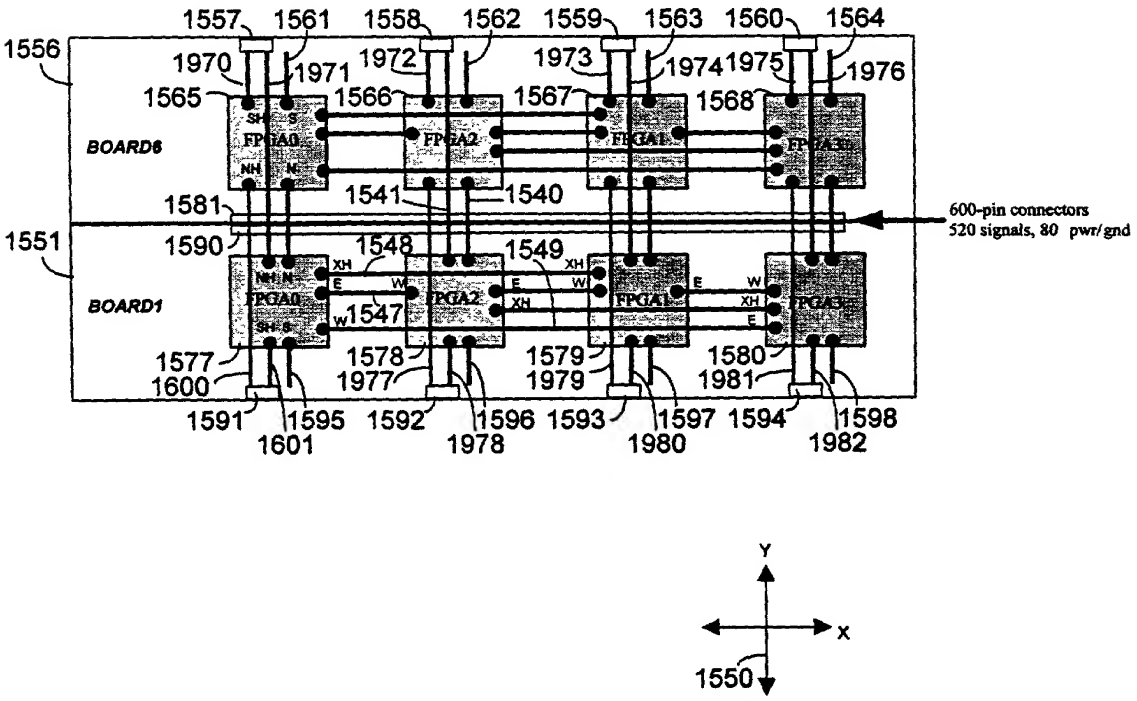


FIG. 44

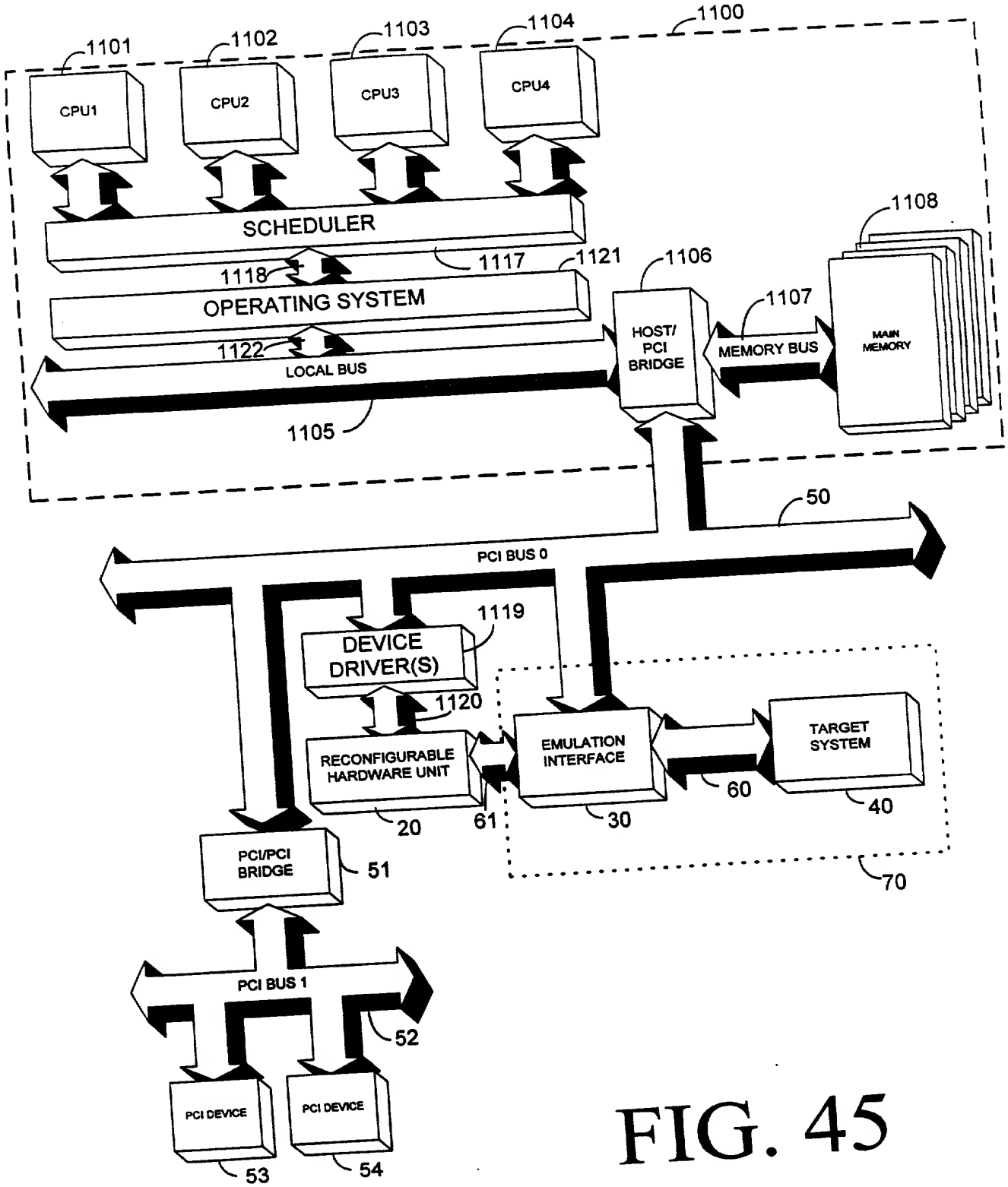

1100
1101
1102
1103
1104
CPU1
CPU2
CPU3
CPU4
SCHEDULER
1118
1117
1121
1106
1108
OPERATING SYSTEM
1107
1122
HOST/
PCI
BRIDGE
MEMORY BUS
MAIN
MEMORY
LOCAL BUS
1105
50
PCI BUS 0
1119
DEVICE
DRIVER(S)
1120
RECONFIGURABLE
HARDWARE UNIT
EMULATION
INTERFACE
TARGET
SYSTEM
60
40
20
61
30
70
PCI/PCI
BRIDGE
51
PCI BUS 1
52
PCI DEVICE
PCI DEVICE
53
54

FIG. 45

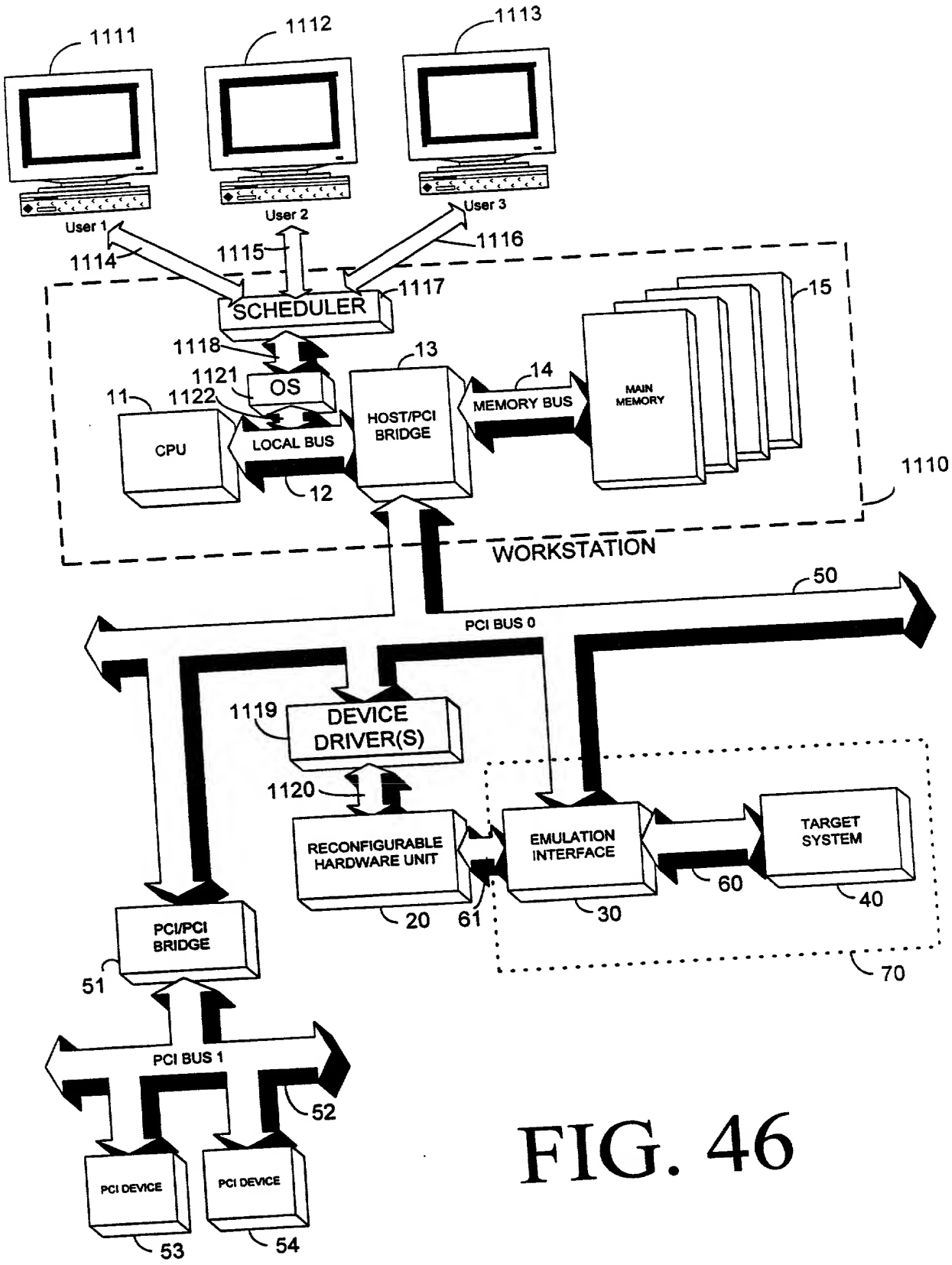

1111
1112
1113
User 1
User 2
User 3
1114
1115
1116
SCHEDULER
1117
1118
1121
OS
1122
11
CPU
LOCAL BUS
HOST/PCI BRIDGE
13
14
MEMORY BUS
MAIN MEMORY
15
12
1110
WORKSTATION
50
PCI BUS 0
1119
DEVICE DRIVER(S)
1120
RECONFIGURABLE HARDWARE UNIT
20
61
EMULATION INTERFACE
30
60
TARGET SYSTEM
40
70
PCI/PCI BRIDGE
51
PCI BUS 1
52
PCI DEVICE
PCI DEVICE
53
54

FIG. 46

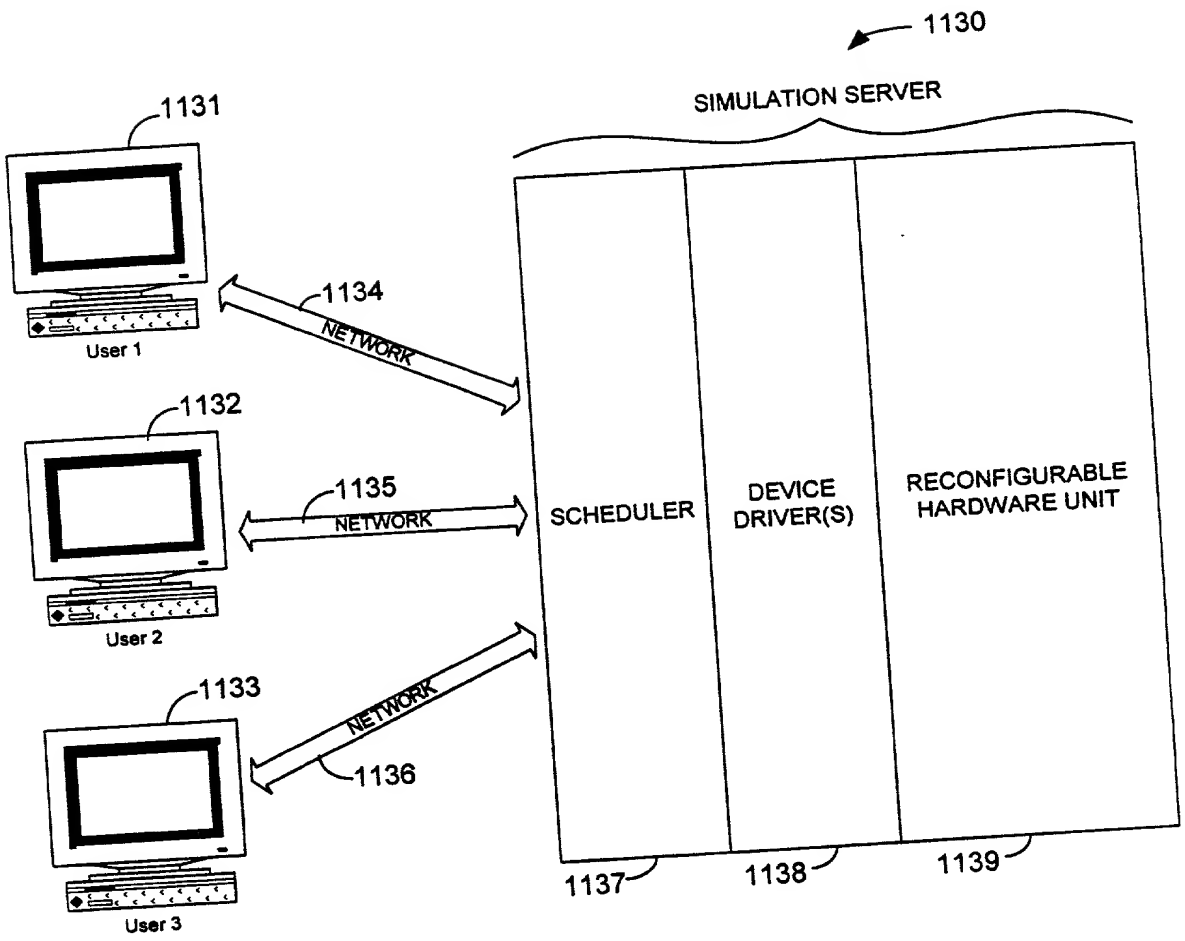

1130
SIMULATION SERVER
1131
1134
NETWORK
User 1
1132
1135
NETWORK
User 2
1133
NETWORK
1136
User 3
SCHEDULER
DEVICE DRIVER(S)
RECONFIGURABLE HARDWARE UNIT
1137
1138
1139

FIG. 47

SIMULATION SERVER ARCHITECTURE

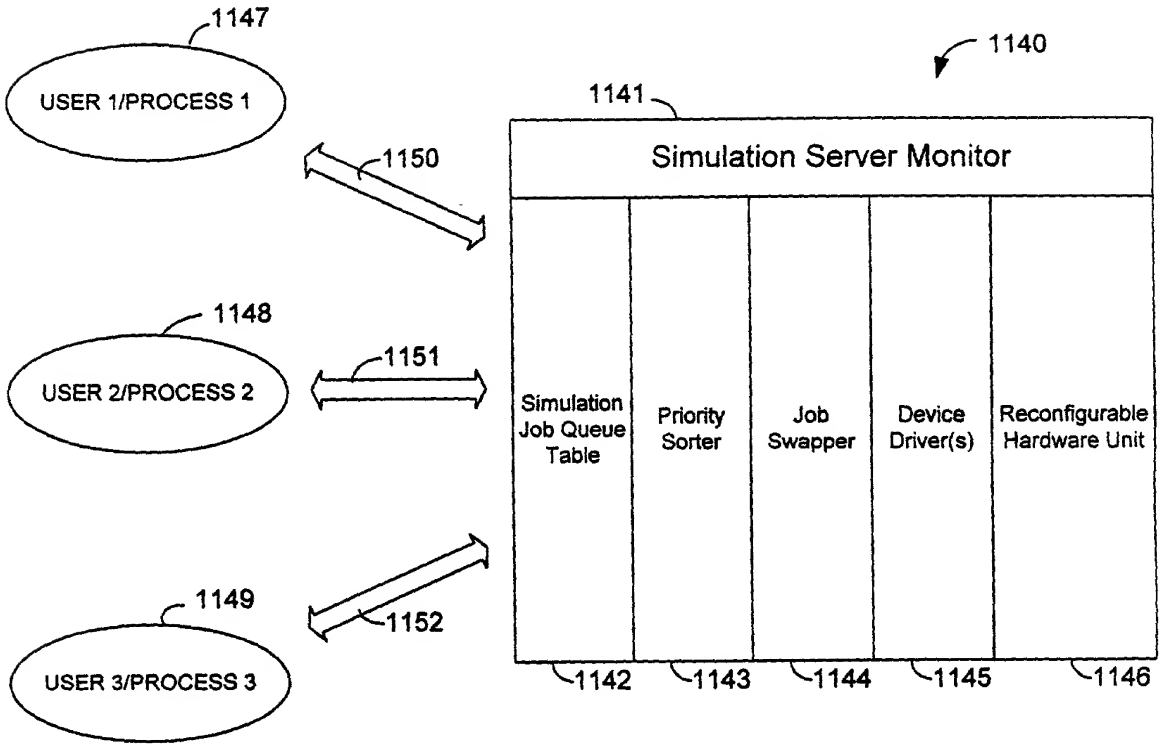


FIG. 48

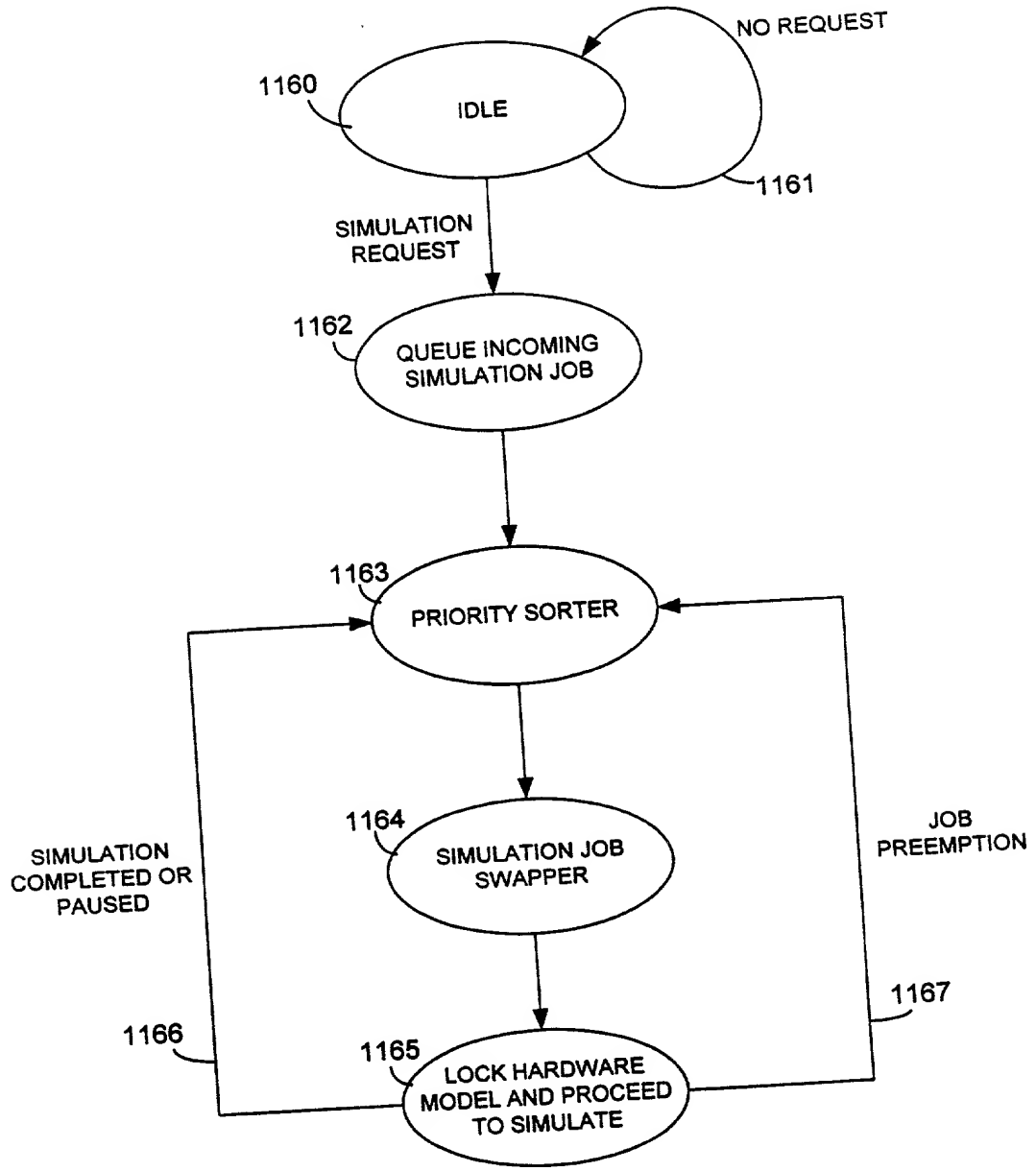

NO REQUEST
1160
IDLE
1161
SIMULATION REQUEST
1162
QUEUE INCOMING SIMULATION JOB
1163
PRIORITY SORTER
1164
SIMULATION JOB SWAPPER
SIMULATION COMPLETED OR PAUSED
JOB PREEMPTION
1167
1166
1165
LOCK HARDWARE MODEL AND PROCEED TO SIMULATE

FIG. 49

JOB SWAPPER

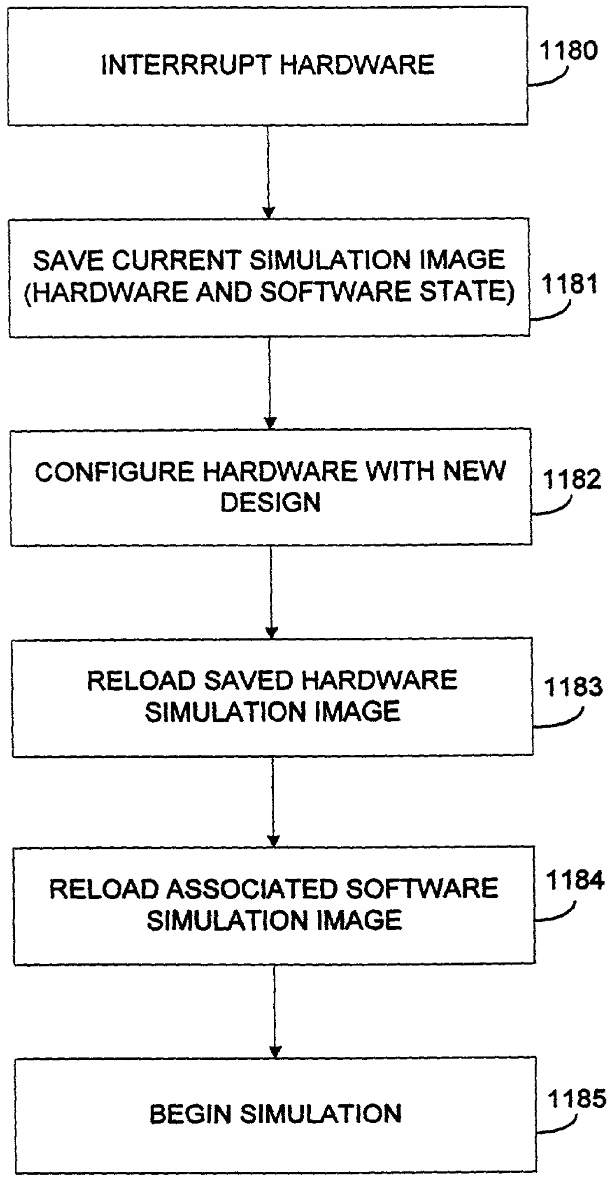


FIG. 50

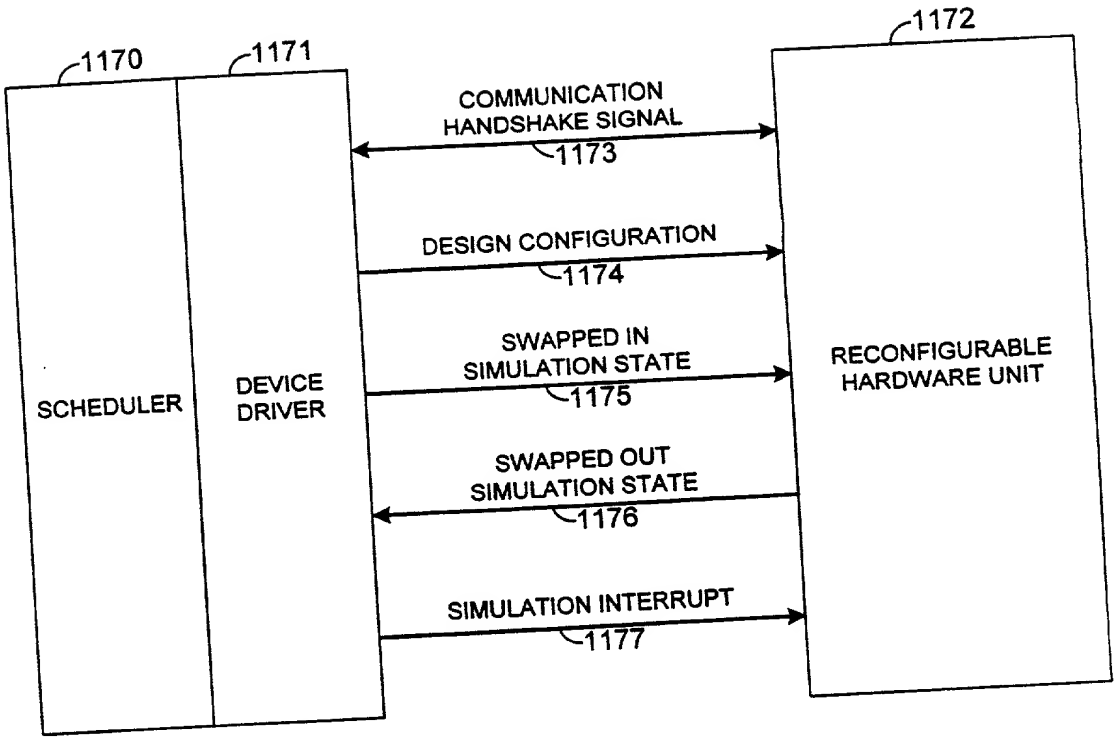

1170
1171
1172
SCHEDULER
DEVICE DRIVER
RECONFIGURABLE HARDWARE UNIT
COMMUNICATION HANDSHAKE SIGNAL
1173
DESIGN CONFIGURATION
1174
SWAPPED IN SIMULATION STATE
1175
SWAPPED OUT SIMULATION STATE
1176
SIMULATION INTERRUPT
1177

FIG. 51

PRIORITY I { JOB A, JOB B

PRIORITY II { JOB C, JOB D

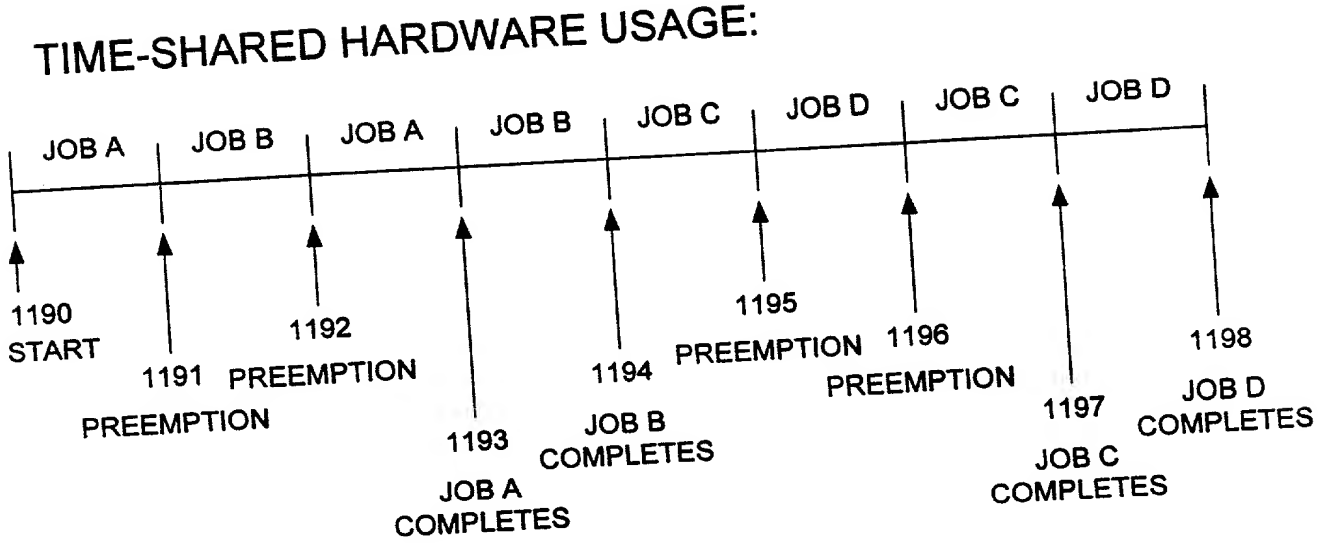


FIG. 52

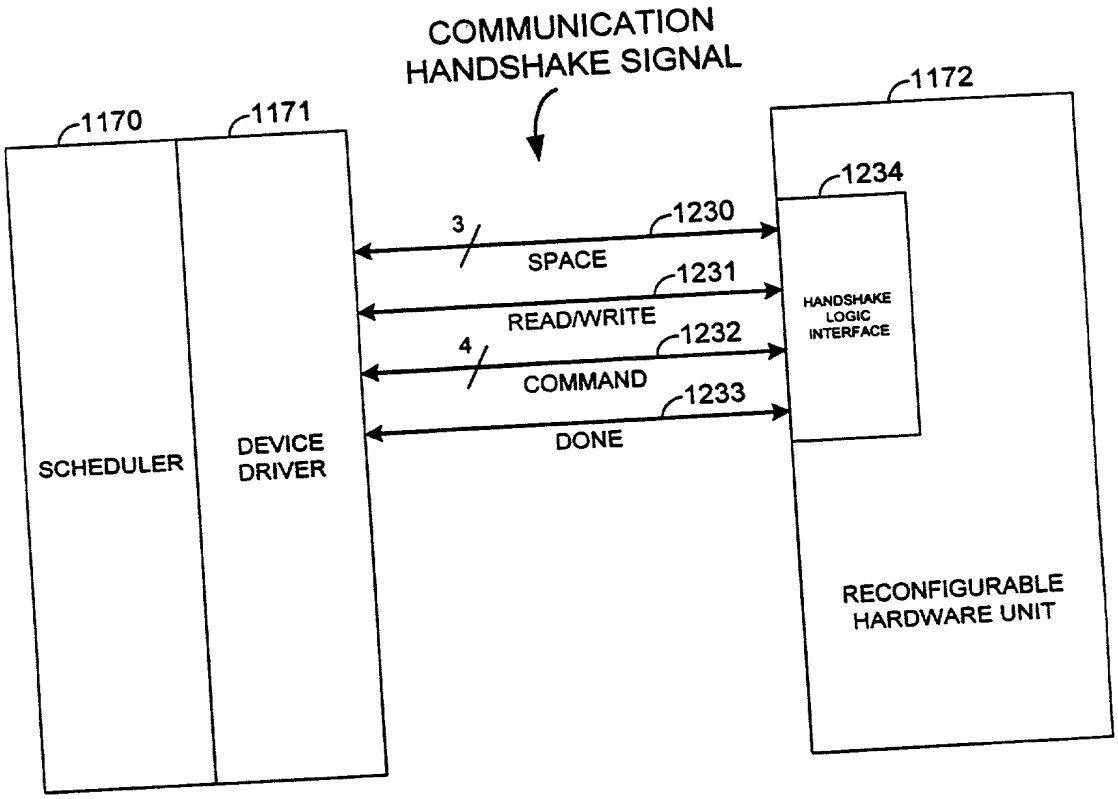

COMMUNICATION
HANDSHAKE SIGNAL
1170
1171
1172
1234
1230
3
SPACE
1231
READ/WRITE
1232
4
COMMAND
1233
DONE
HANDSHAKE
LOGIC
INTERFACE
SCHEDULER
DEVICE
DRIVER
RECONFIGURABLE
HARDWARE UNIT

FIG. 53

COMMUNICATION HANDSHAKE PROTOCOL

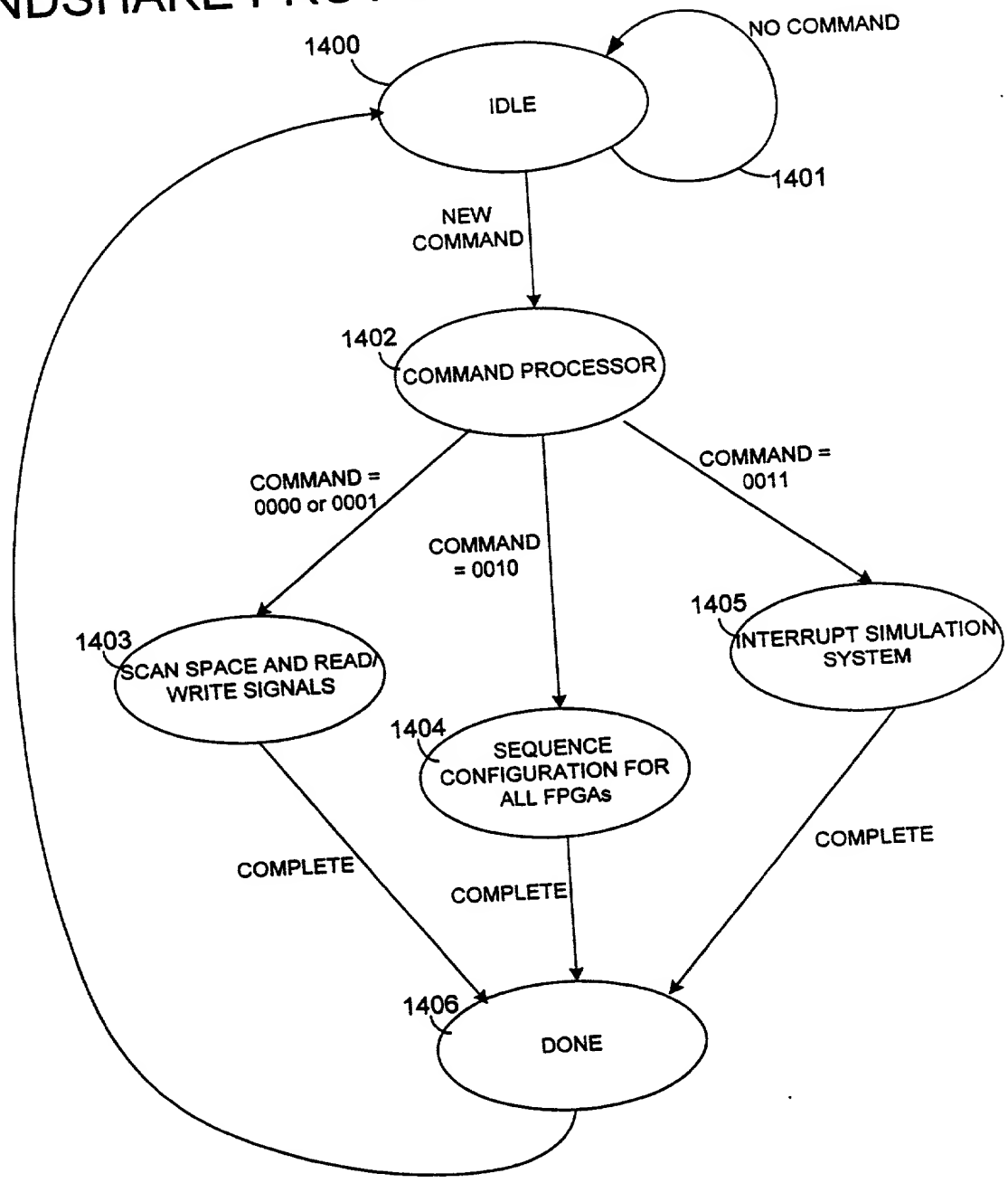


FIG. 54

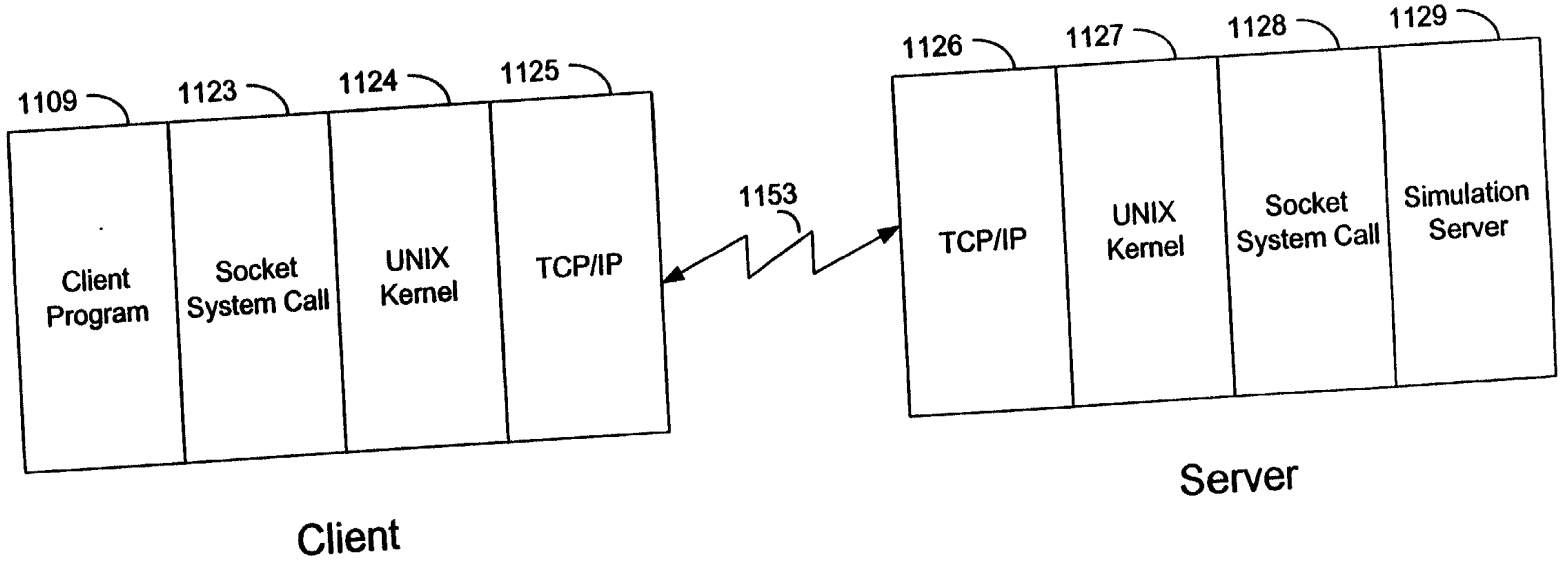

1109
1123
1124
1125
Client Program
Socket System Call
UNIX Kernel
TCP/IP
Client
1153
1126
1127
1128
1129
TCP/IP
UNIX Kernel
Socket System Call
Simulation Server
Server

FIG. 55

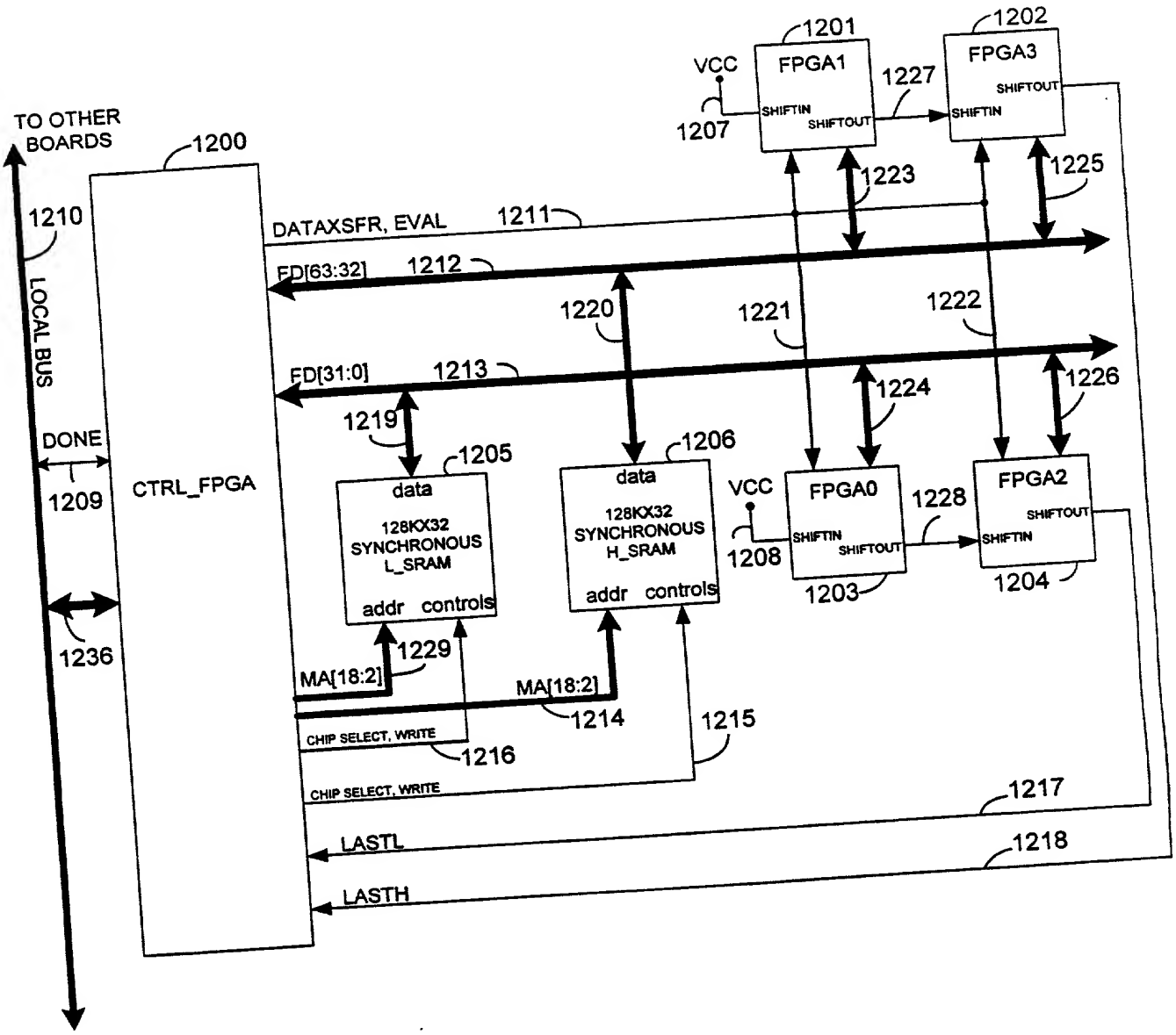

TO OTHER BOARDS
1200
1210
LOCAL BUS
DONE
1209
CTRL_FPGA
1236
DATAXSFR, EVAL
1211
FD[63:32]
1212
FD[31:0]
1213
1219
1205
data
128KX32 SYNCHRONOUS L_SRAM
addr controls
1206
data
128KX32 SYNCHRONOUS H_SRAM
addr controls
MA[18:2]
1229
MA[18:2]
1214
CHIP SELECT, WRITE
1216
CHIP SELECT, WRITE
1215
LASTL
LASTH
1217
1218
VCC
1207
1201
FPGA1
SHIFTIN
SHIFTOUT
1227
1202
FPGA3
SHIFTOUT
SHIFTIN
1223
1225
1220
1221
1222
1224
1226
VCC
1208
FPGA0
SHIFTIN
SHIFTOUT
1203
1228
FPGA2
SHIFTOUT
SHIFTIN
1204

FIG. 56

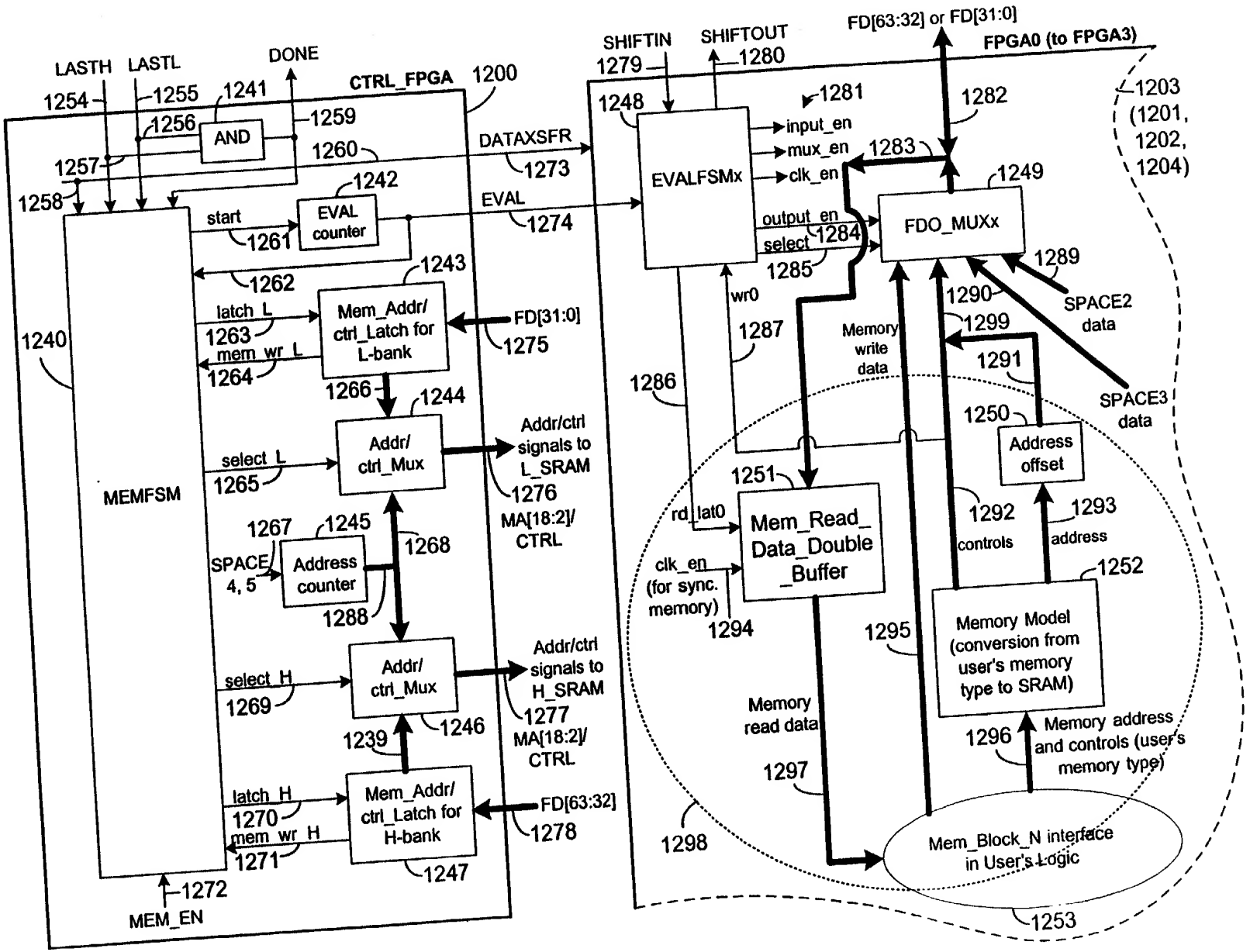

LASTH
LASTL
DONE
CTRL_FPGA
1200
1254
1255
1241
1256
AND
1259
1257
1260
1258
DATAXSFR
1273
1242
start
EVAL counter
1261
EVAL
1274
1262
1243
latch_L
1263
mem_wr_L
1264
Mem_Addr/ ctrl_Latch for L-bank
FD[31:0]
1275
1240
1266
1244
Addr/ ctrl_Mux
Addr/ctrl signals to L_SRAM
1276
MA[18:2]/ CTRL
select_L
1265
MEMFSM
1267
1245
SPACE 4, 5
Address counter
1268
1288
Addr/ ctrl_Mux
Addr/ctrl signals to H_SRAM
1277
MA[18:2]/ CTRL
select_H
1269
1246
1239
latch_H
1270
mem_wr_H
1271
Mem_Addr/ ctrl_Latch for H-bank
FD[63:32]
1278
1247
1272
MEM_EN
SHIFTIN
1279
SHIFTOUT
1280
FD[63:32] or FD[31:0]
FPGA0 (to FPGA3)
1203 (1201, 1202, 1204)
1248
1281
1282
EVALFSMx
input_en
mux_en
clk_en
1283
1249
output_en
select
1284
1285
FDO_MUXx
1289
wr0
1290
1299
SPACE2 data
1287
Memory write data
1291
1286
SPACE3 data
1250
Address offset
1251
rd_lat0
Mem_Read_ Data_Double _Buffer
1292
controls
1293
address
clk_en (for sync. memory)
1294
1252
1295
Memory Model (conversion from user's memory type to SRAM)
Memory read data
1296
Memory address and controls (user's memory type)
1297
1298
Mem_Block_N interface in User's Logic
1253

FIG. 57

MEMFSM - Memory Finite State Machine in CTRL_FPGA unit

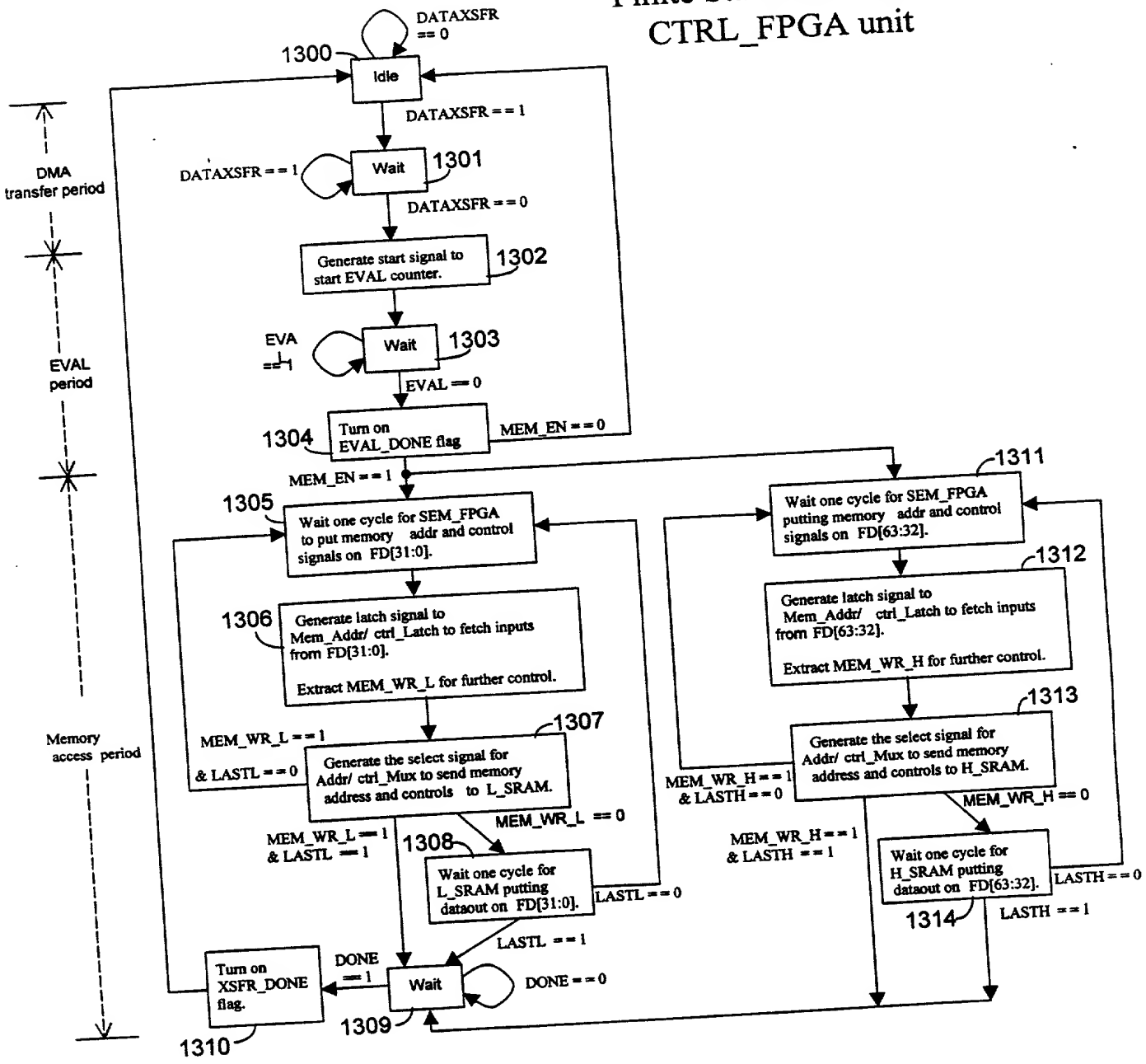


FIG. 58

EVALFSM - EVAL
Finite State Machine in
each FPGA logic device

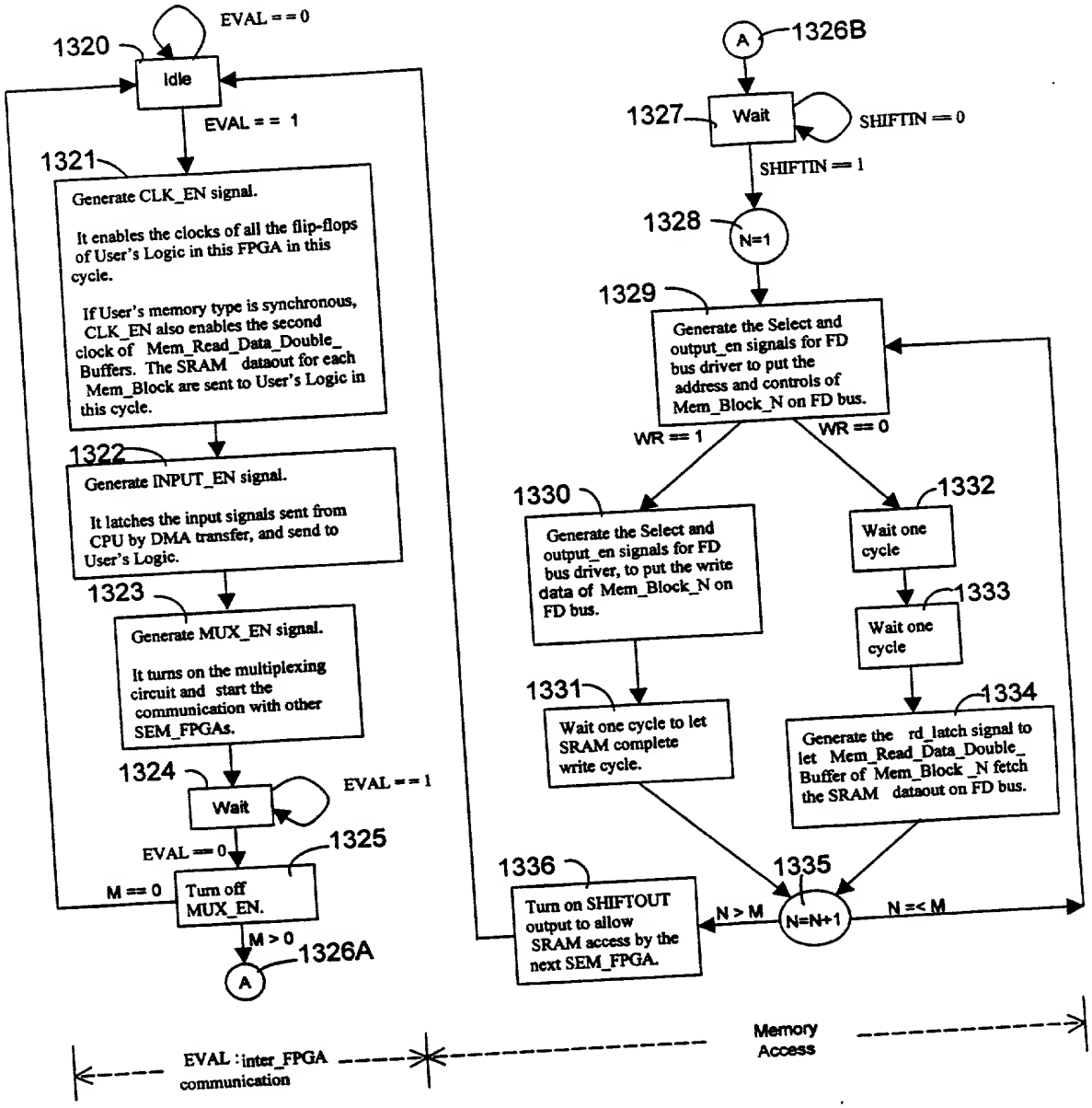


FIG. 59

MEMORY READ DATA DOUBLE BUFFER

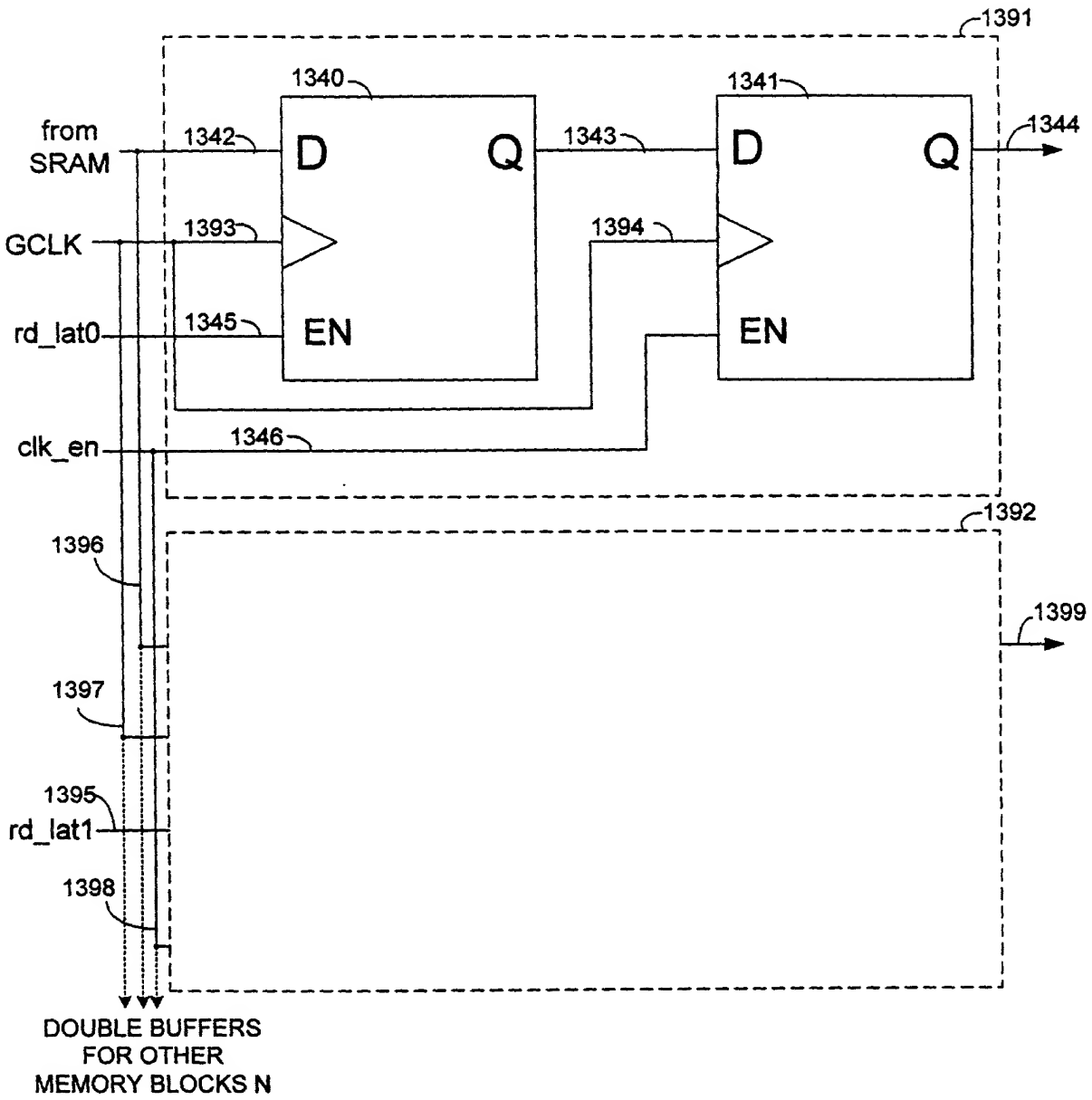


FIG. 60

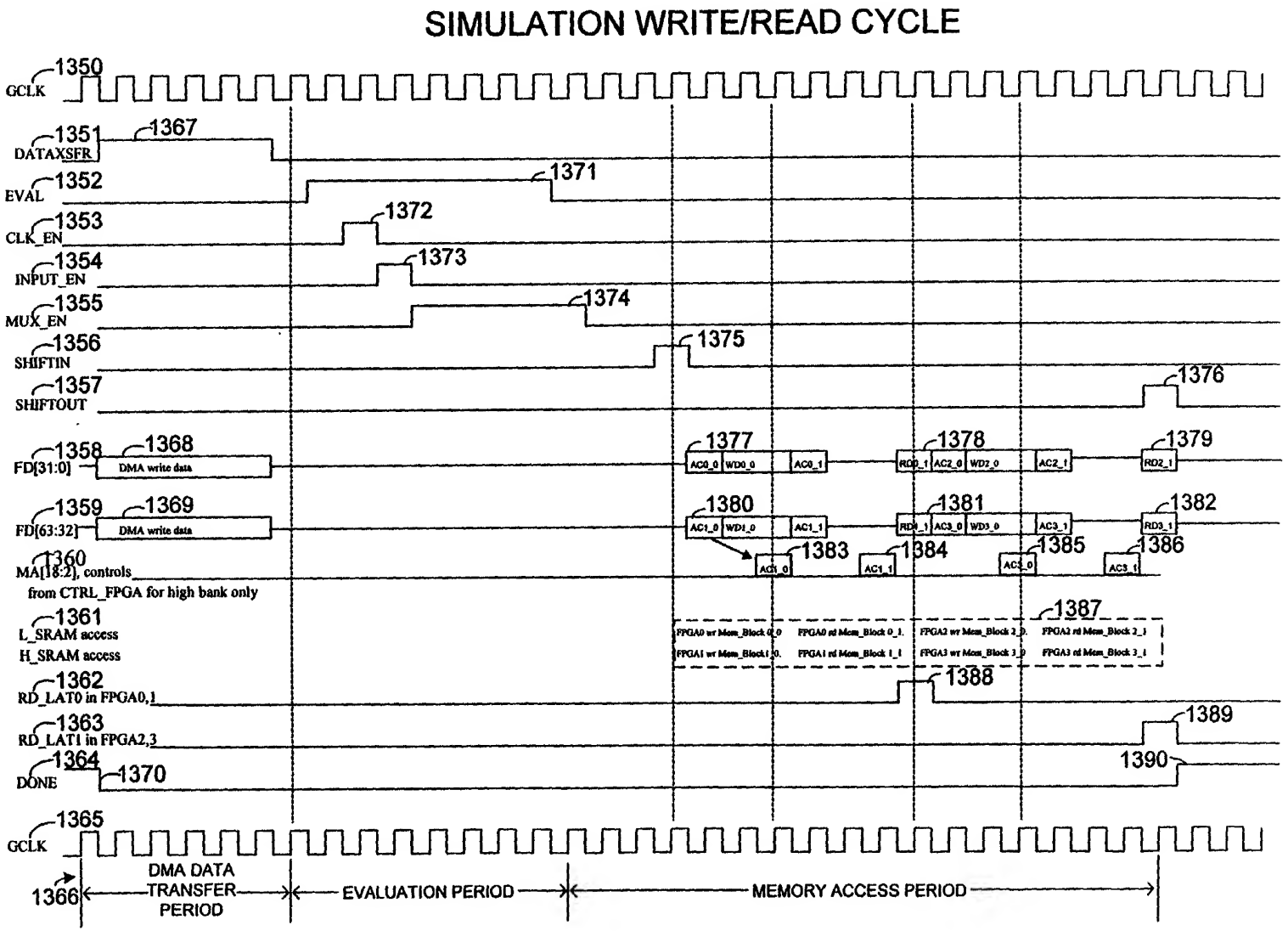


FIG. 61

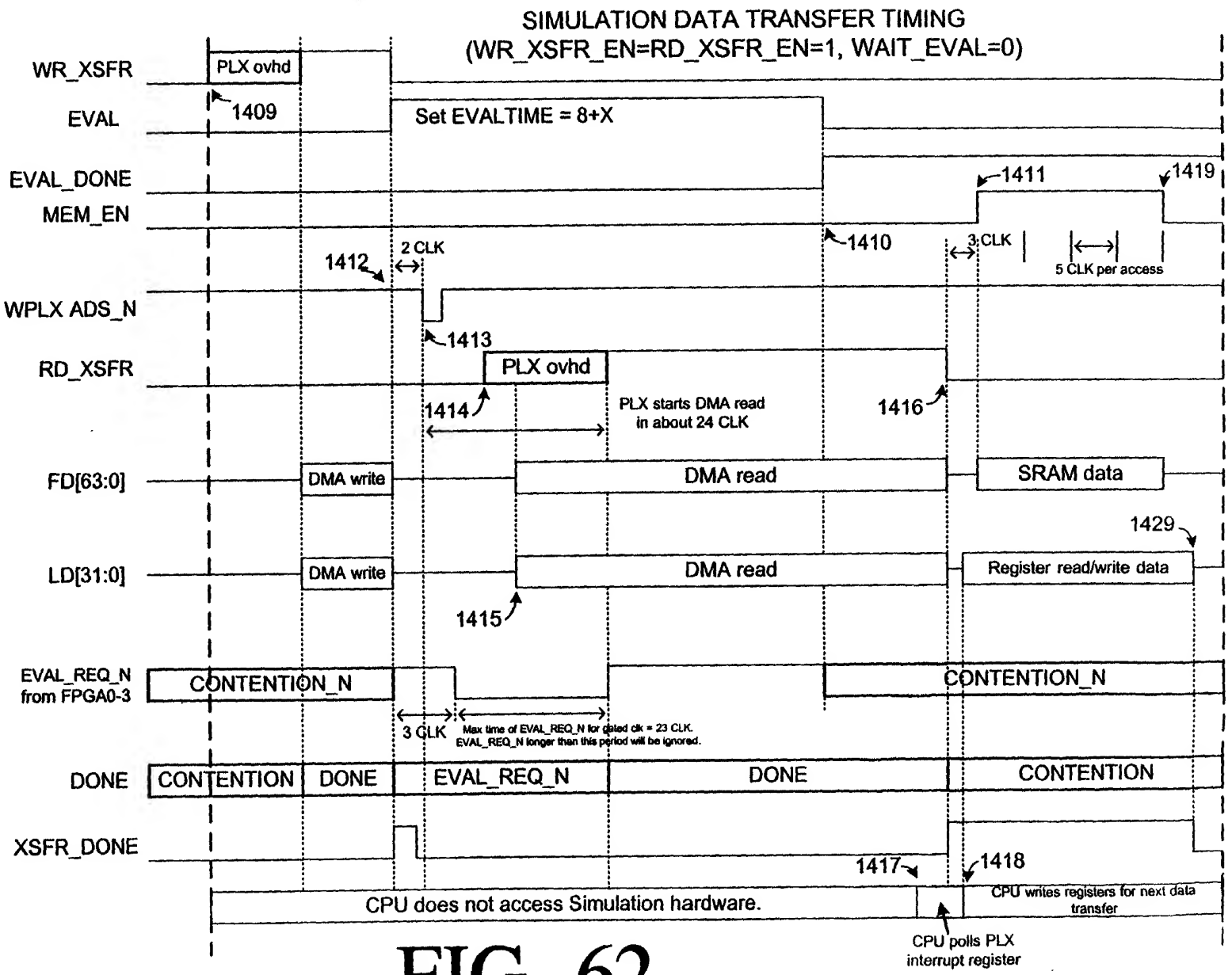


FIG. 62

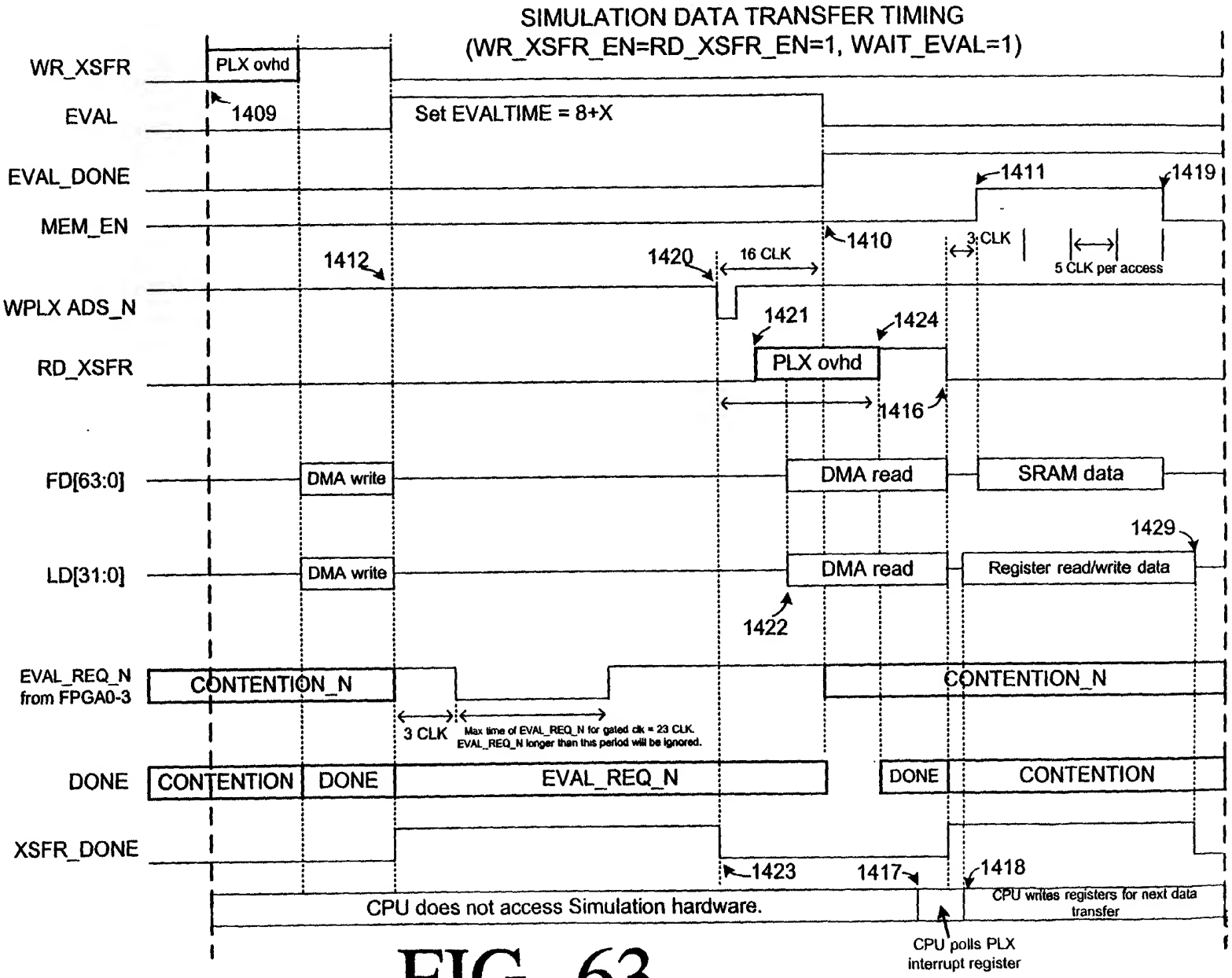


FIG. 63

Typical User Design of PCI Add-on Cards

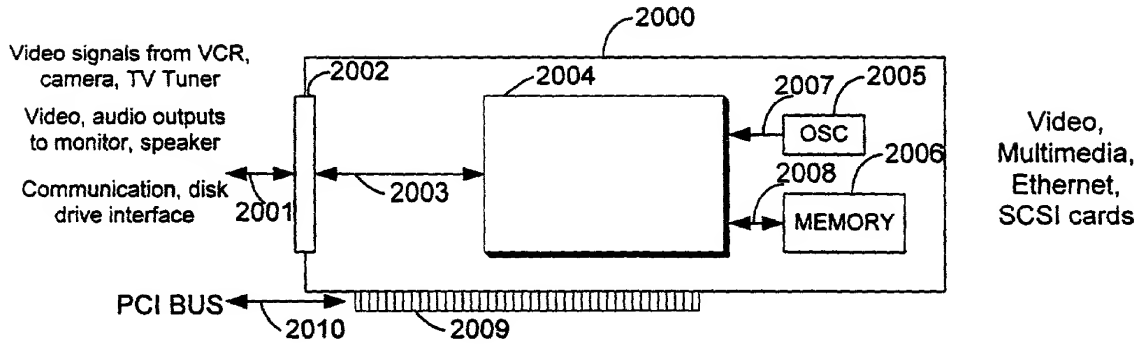


FIG. 64

Typical Hardware/Software Co-Verification

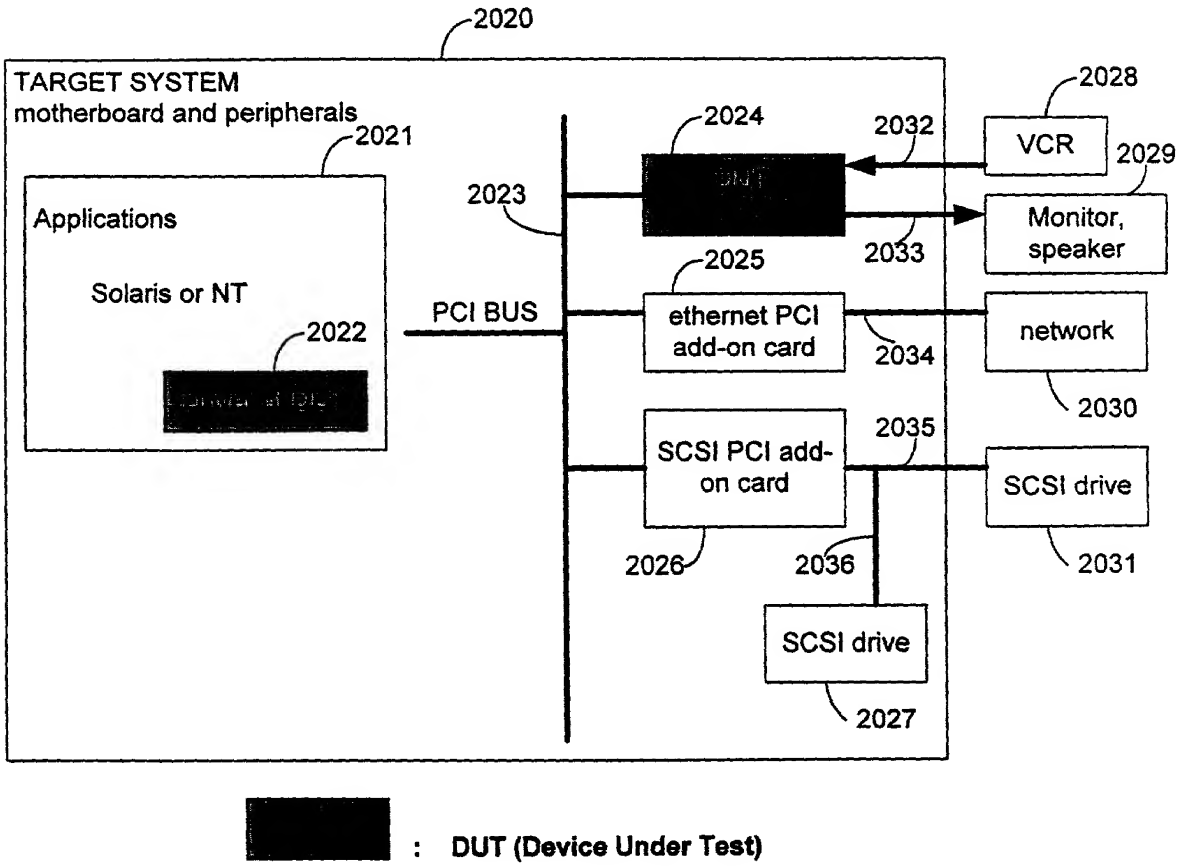
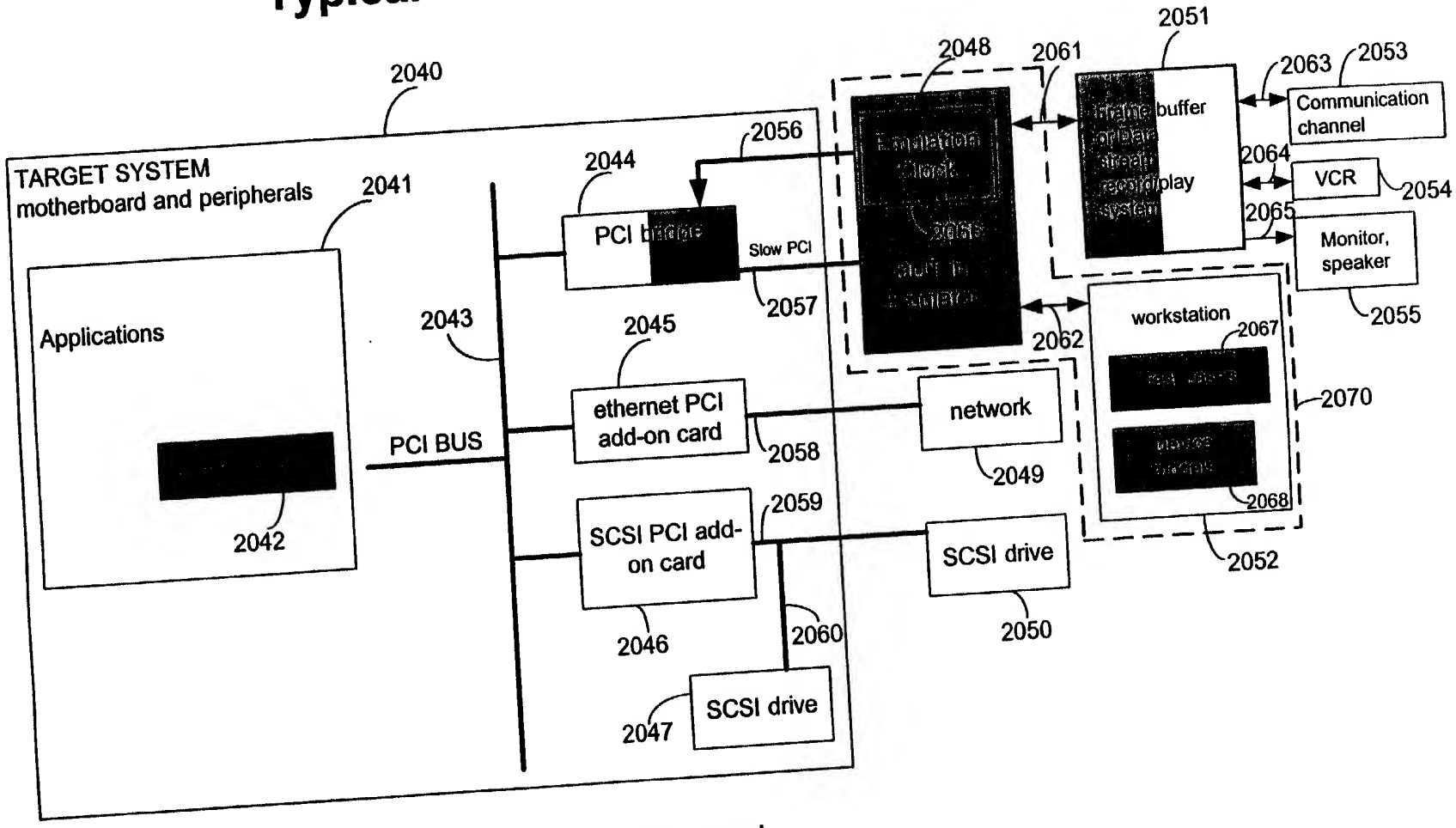


FIG. 65

Typical Co-Verification by Using Emulator



: running time at emulation speed

The rest of the target system is running at full speed.

FIG. 66

SIMULATION

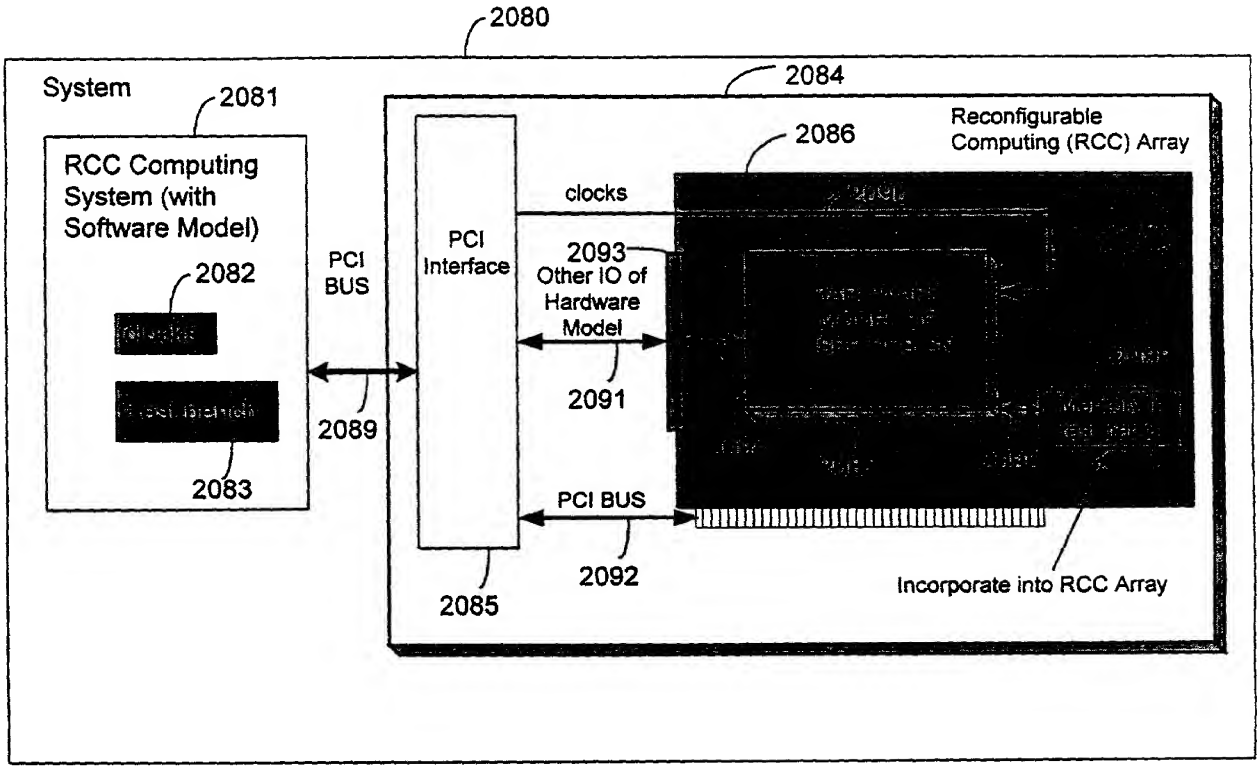


FIG. 67

CO-VERIFICATION WITHOUT EXTERNAL I/O

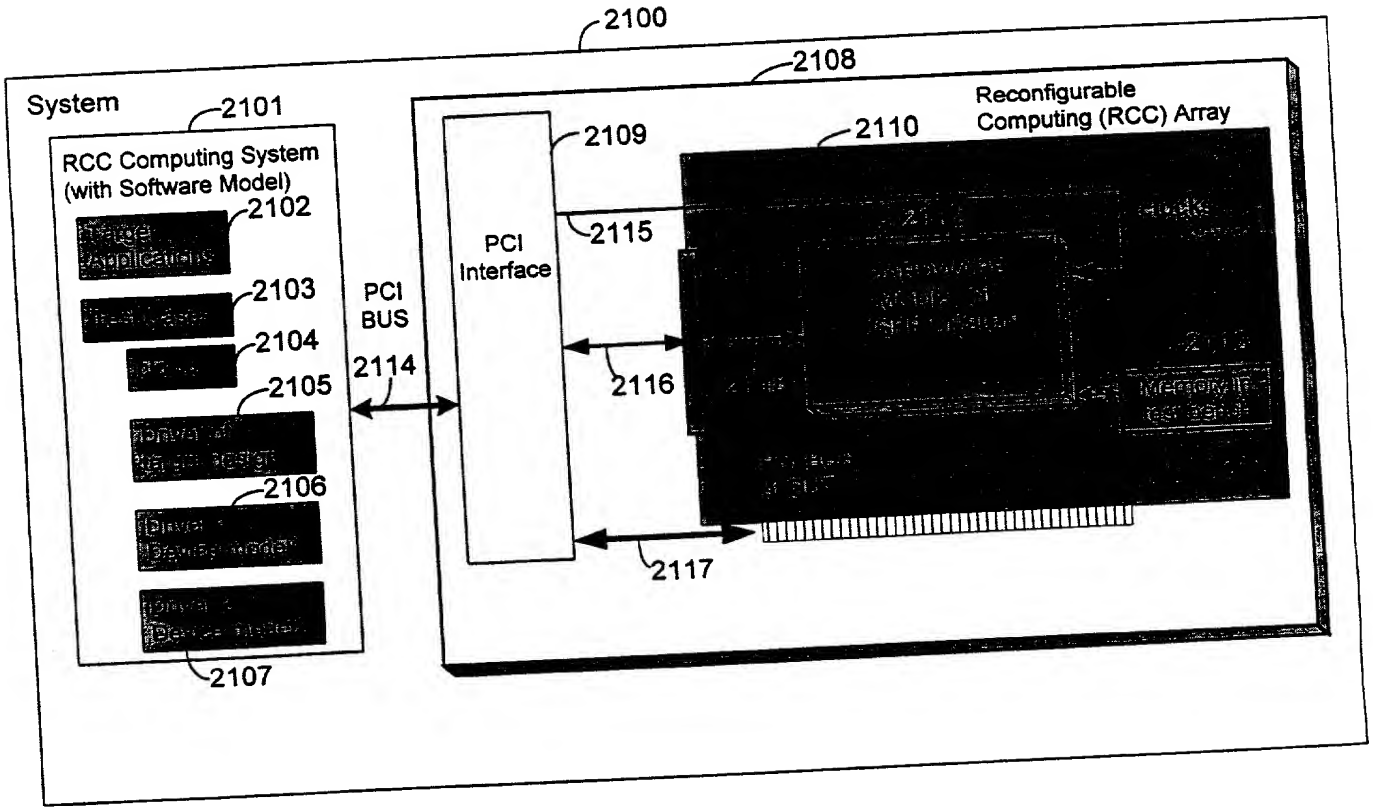


FIG. 68

CO-VERIFICATION WITH EXTERNAL I/O

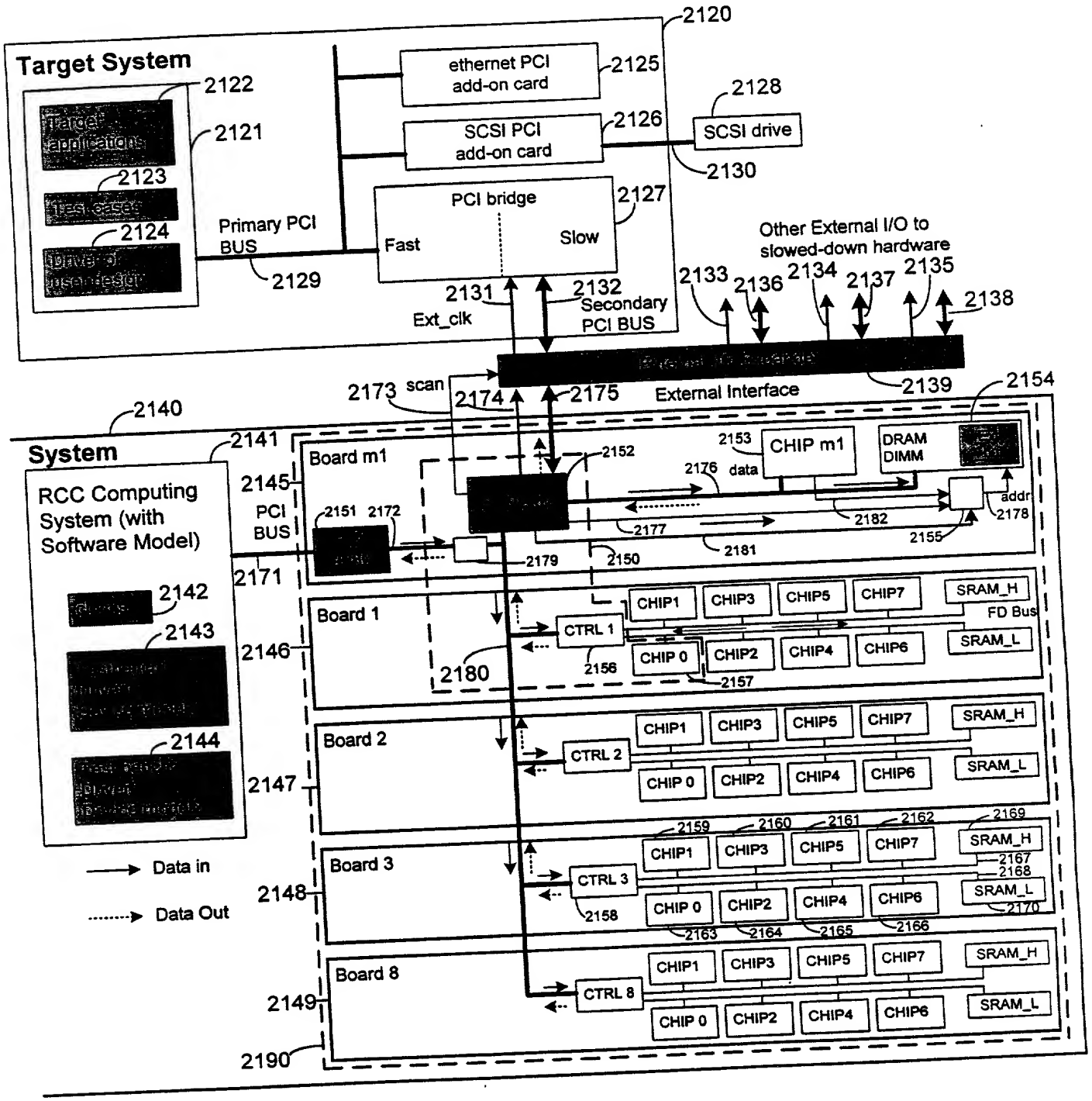


FIG. 69

CONTROL OF DATA-IN CYCLE

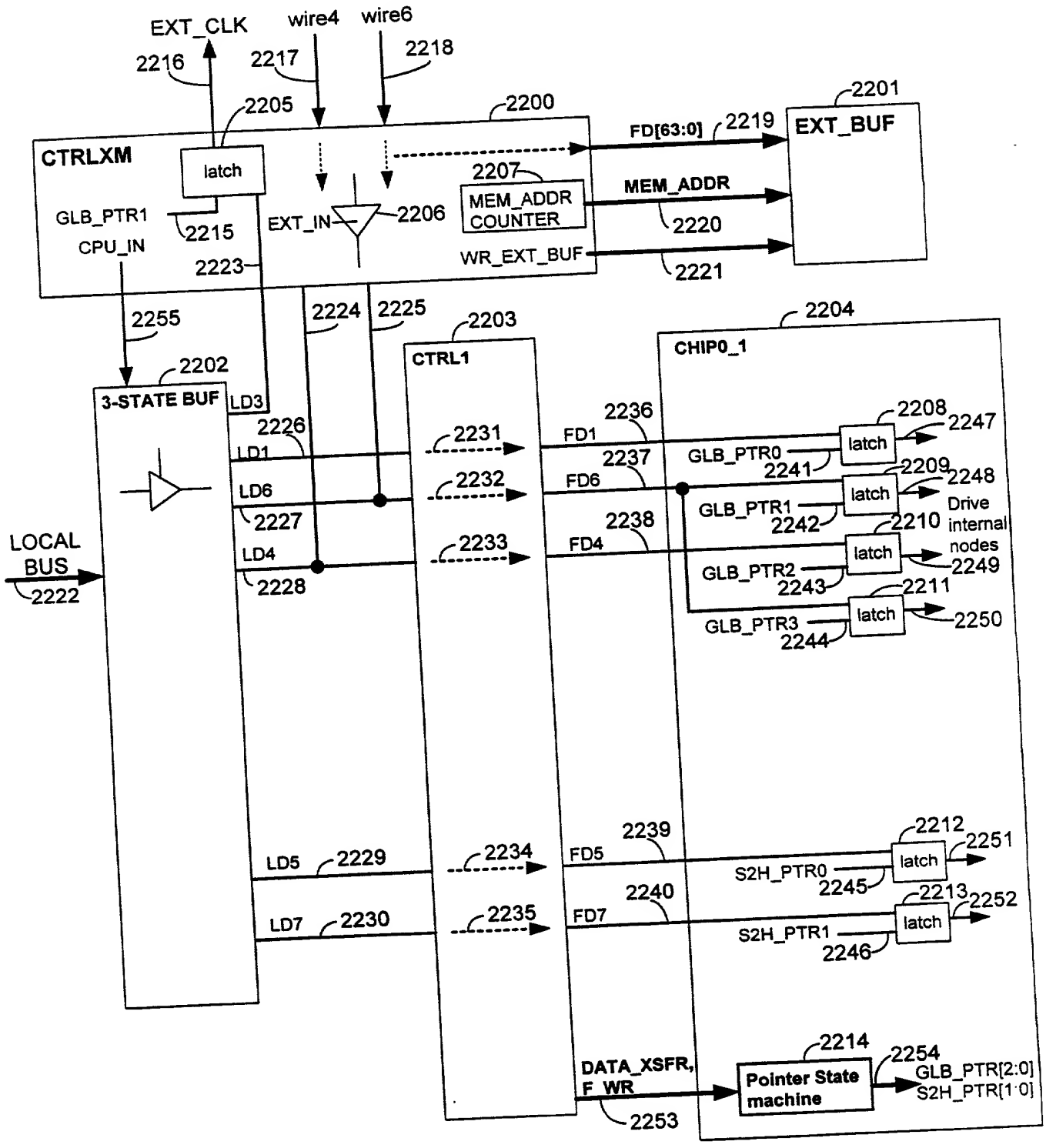


FIG. 70

CONTROL OF DATA-OUT CYCLE

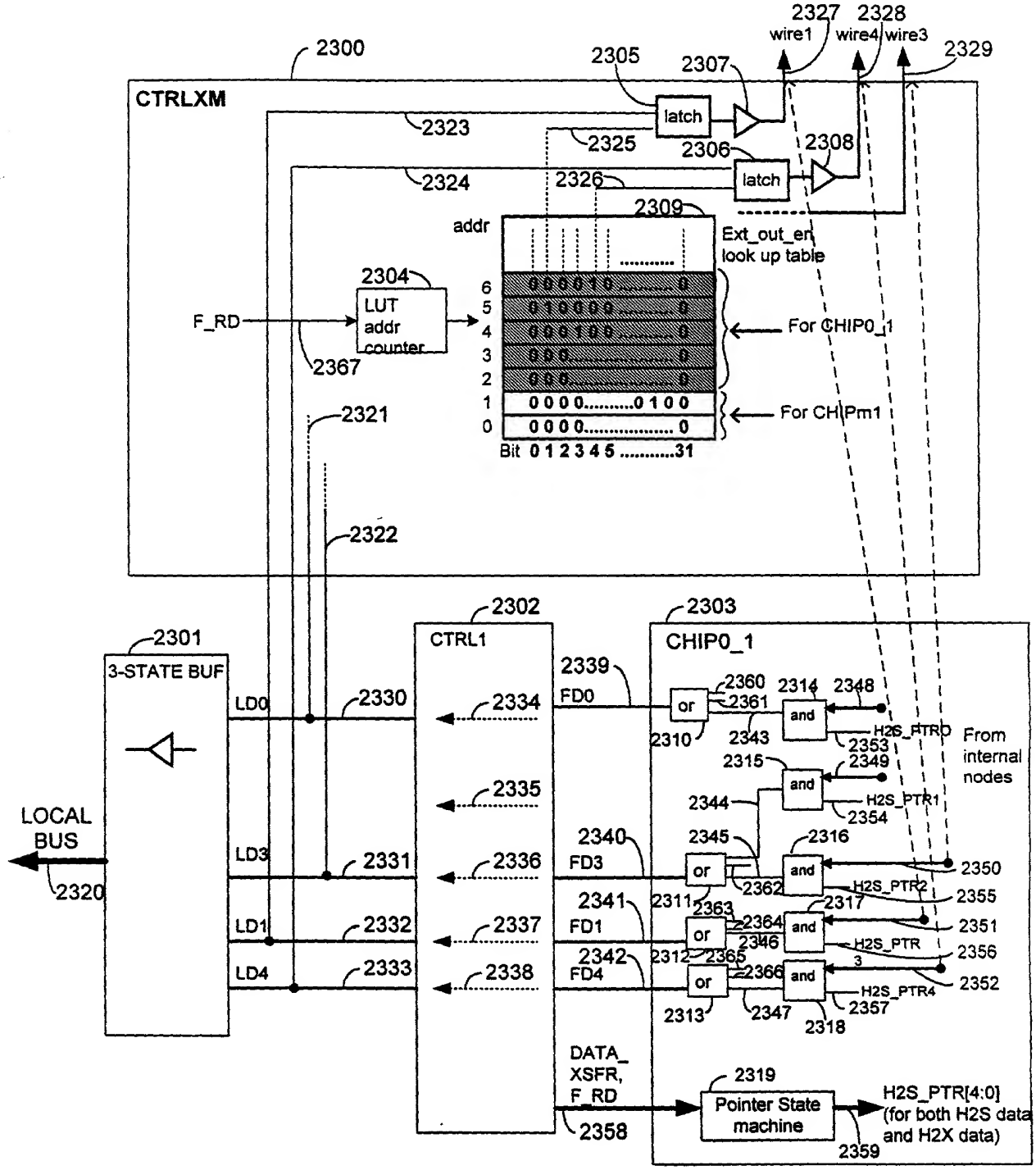


FIG. 71

CONTROL OF DATA-IN CYCLE

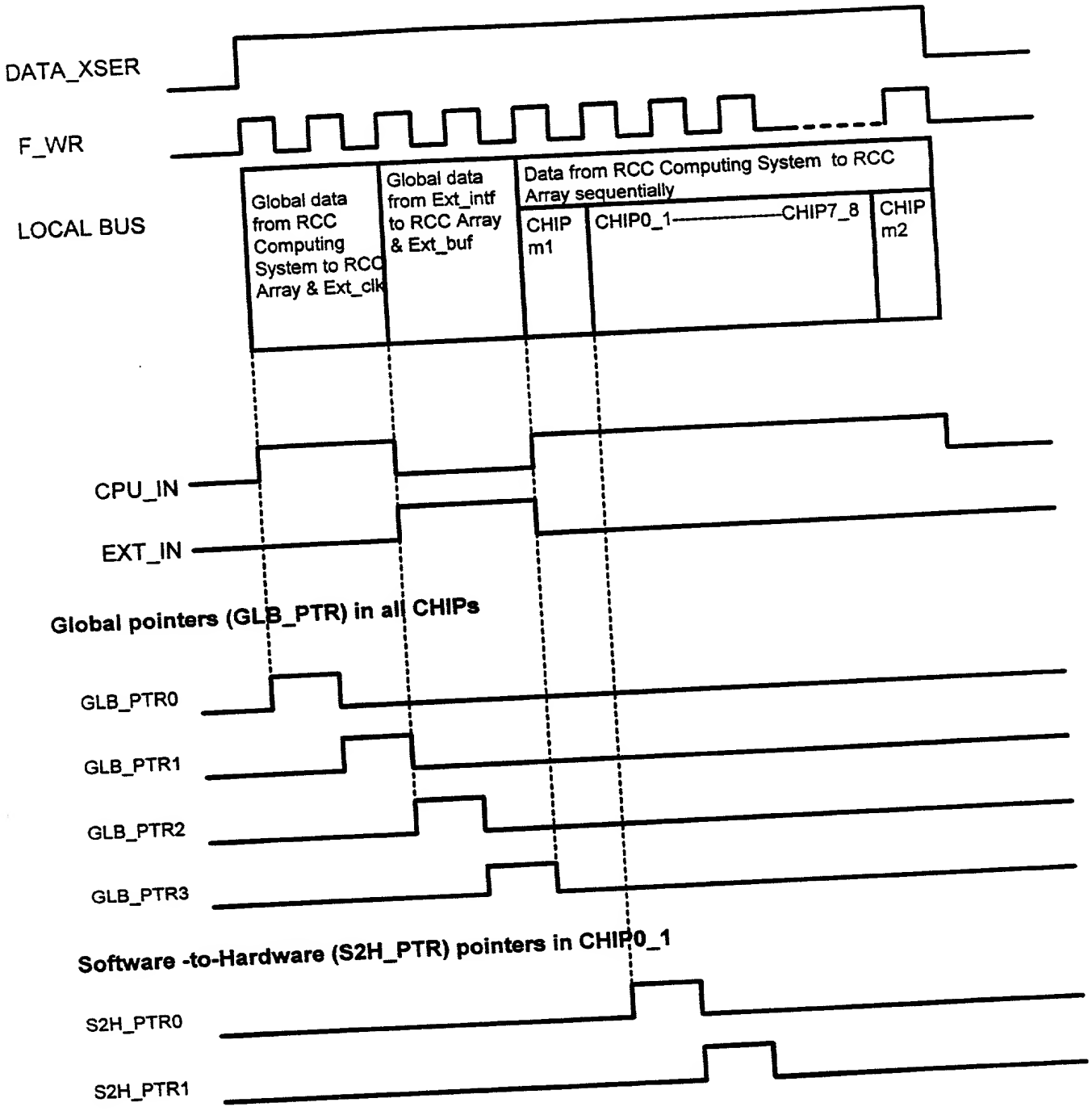


FIG. 72

CONTROL OF DATA-OUT CYCLE

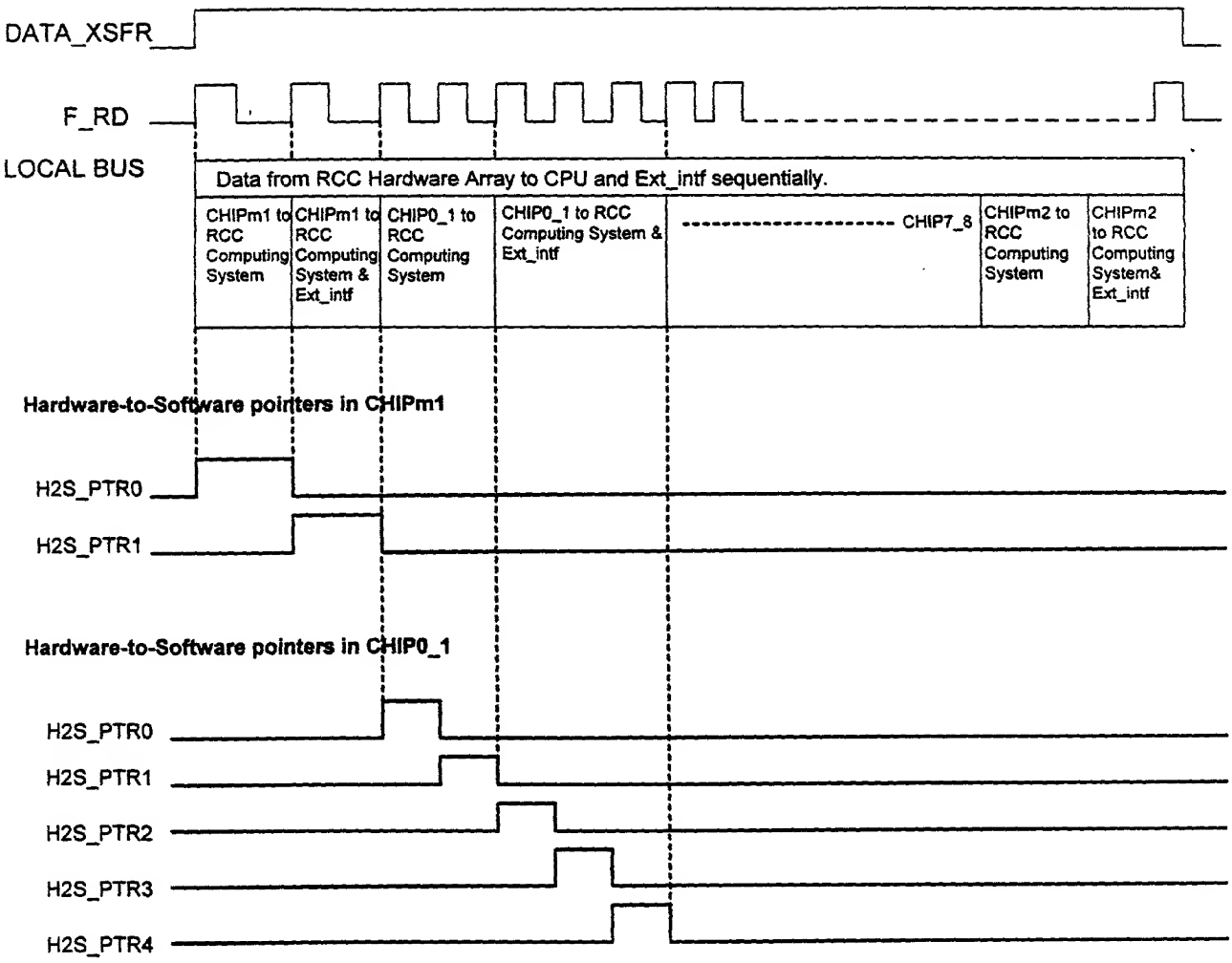


FIG. 73

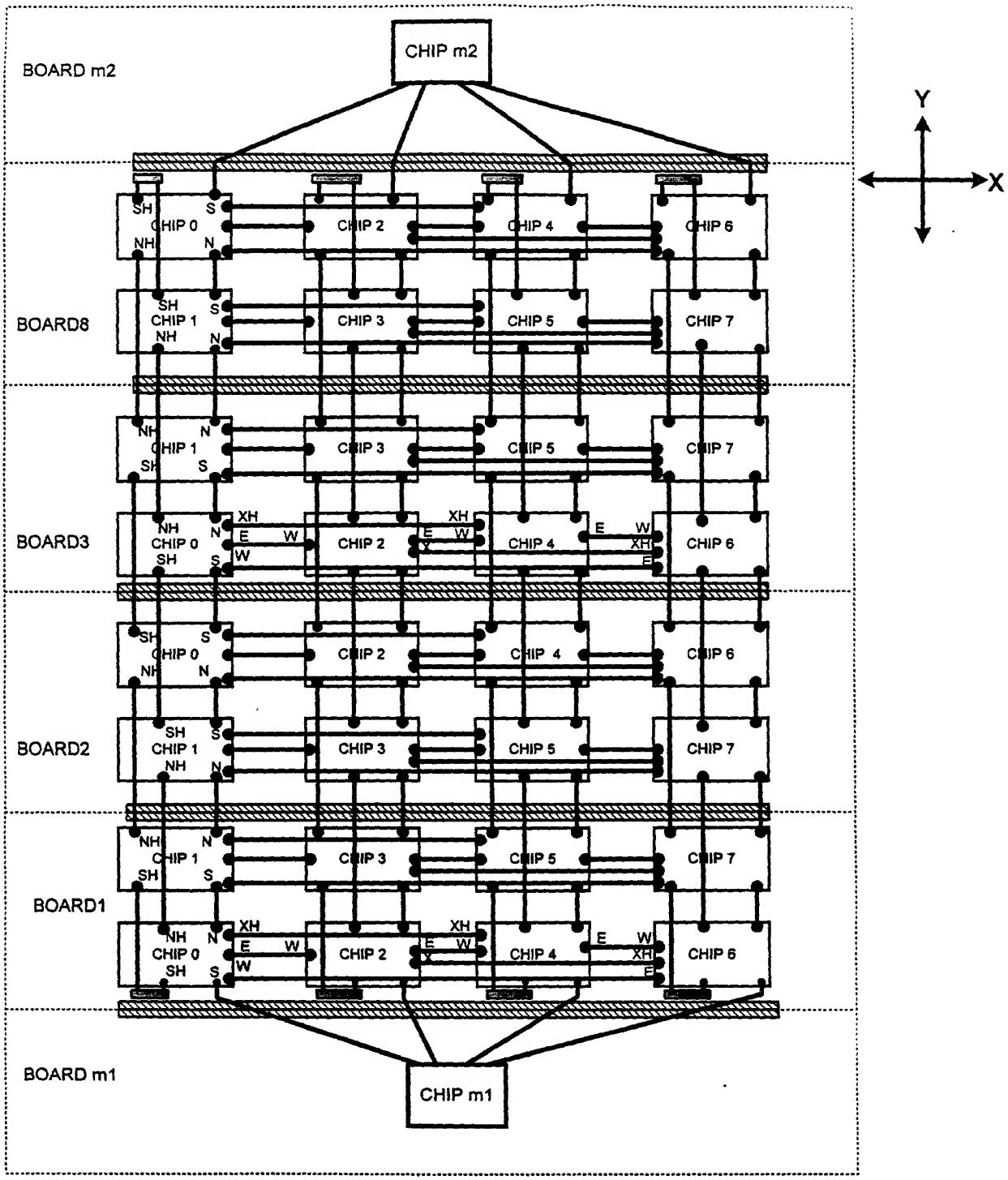

BOARD m2
CHIP m2
Y
X
BOARD8
BOARD3
BOARD2
BOARD1
BOARD m1
CHIP m1
CHIP 0
CHIP 1
CHIP 2
CHIP 3
CHIP 4
CHIP 5
CHIP 6
CHIP 7
SH
NH
S
N
XH
E
W

FIG. 74

SHIFT REGISTER

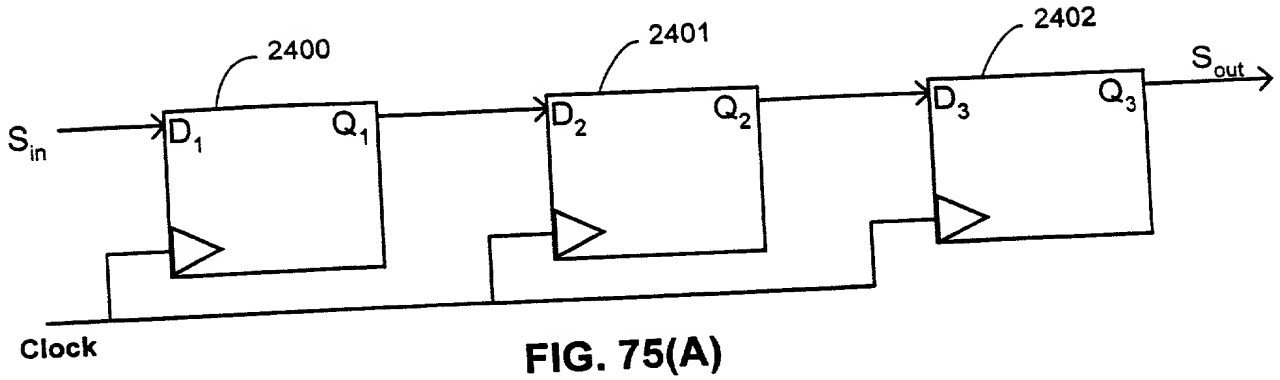


FIG. 75(A)

HOLD TIME ASSUMPTION FOR SHIFT REGISTER

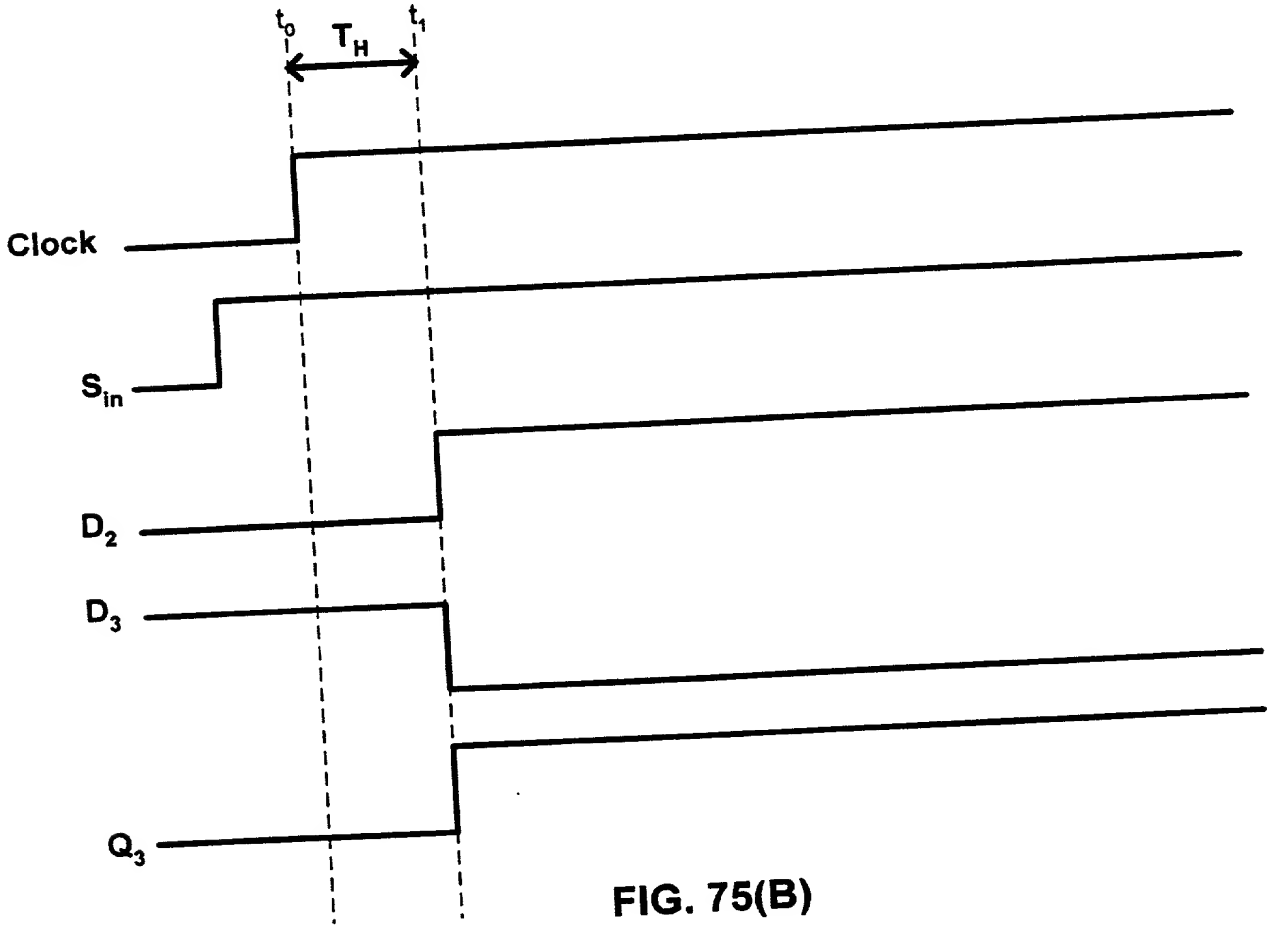


FIG. 75(B)

MULTIPLE FPGA MAPPING FOR SHIFT REGISTER

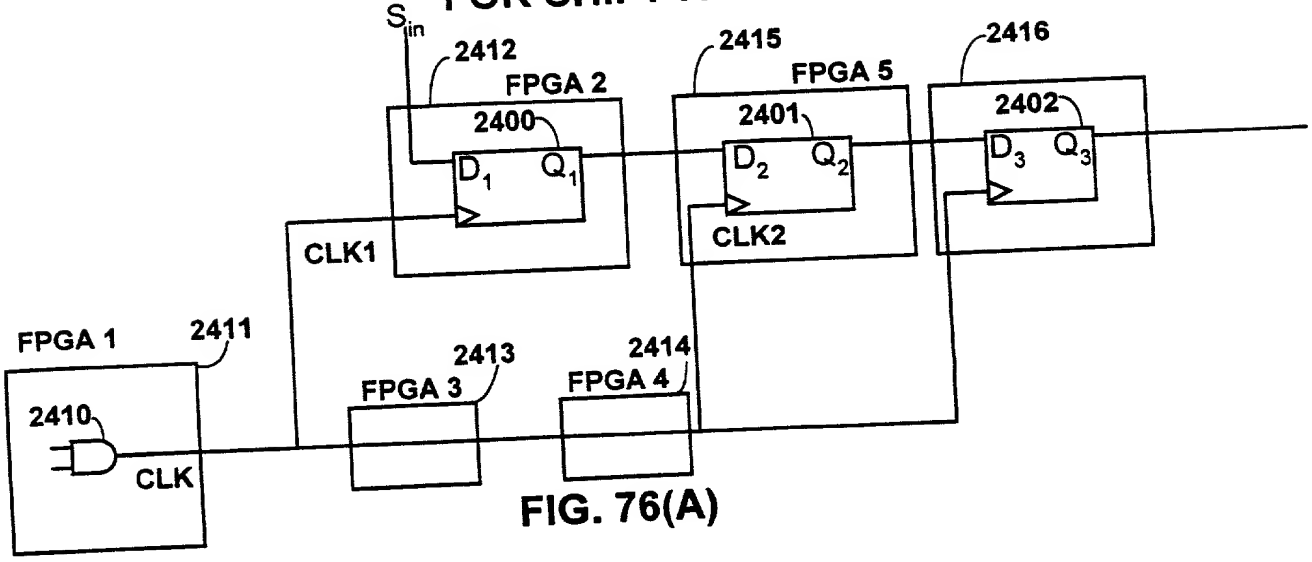


FIG. 76(A)

HOLD TIME VIOLATION BY LONG CLOCK SKEW

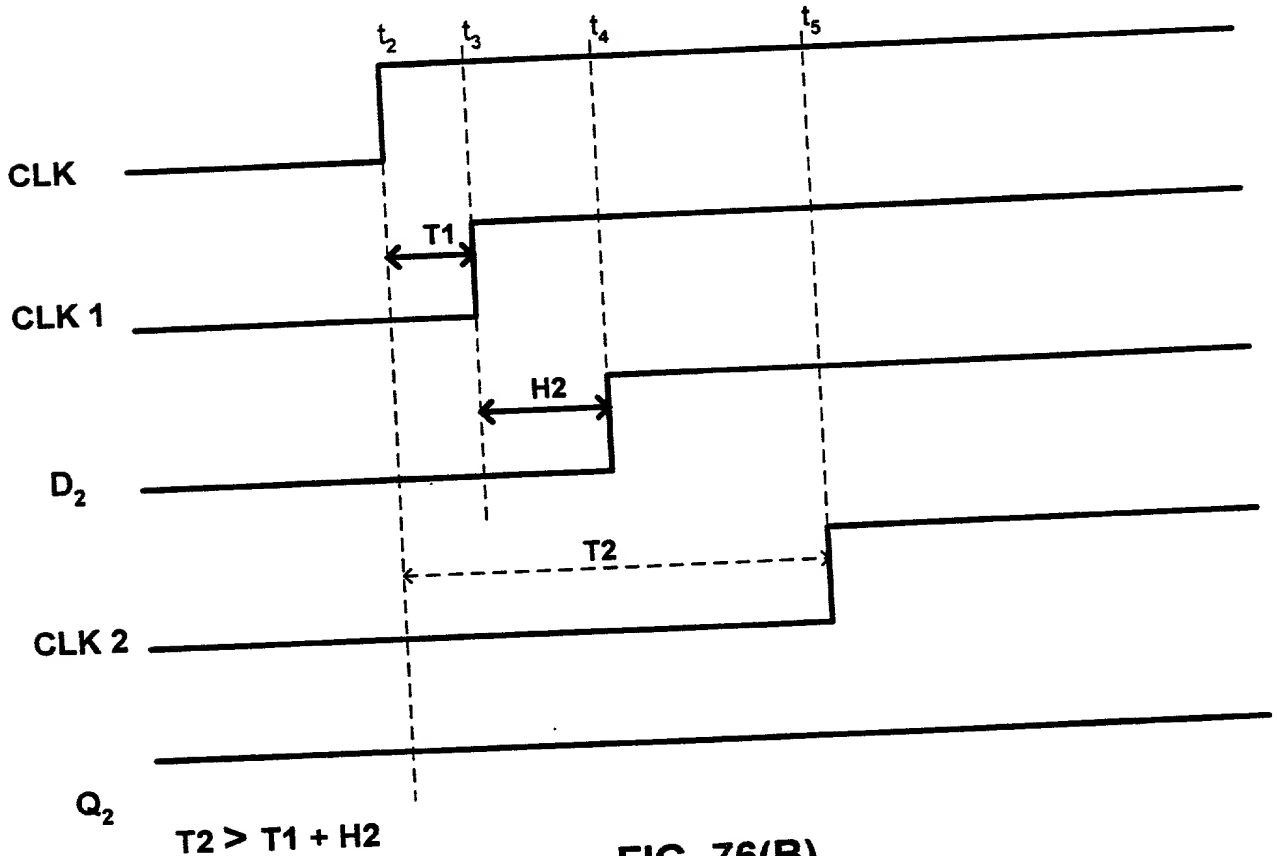


FIG. 76(B)

CLOCK GLITCH PROBLEM

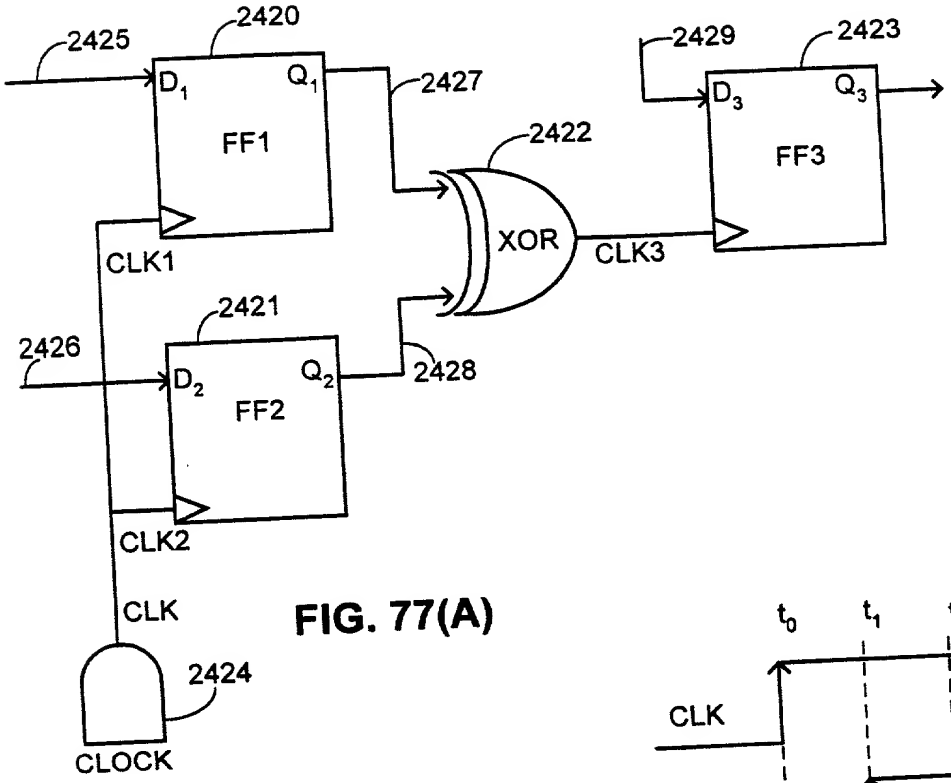


FIG. 77(A)

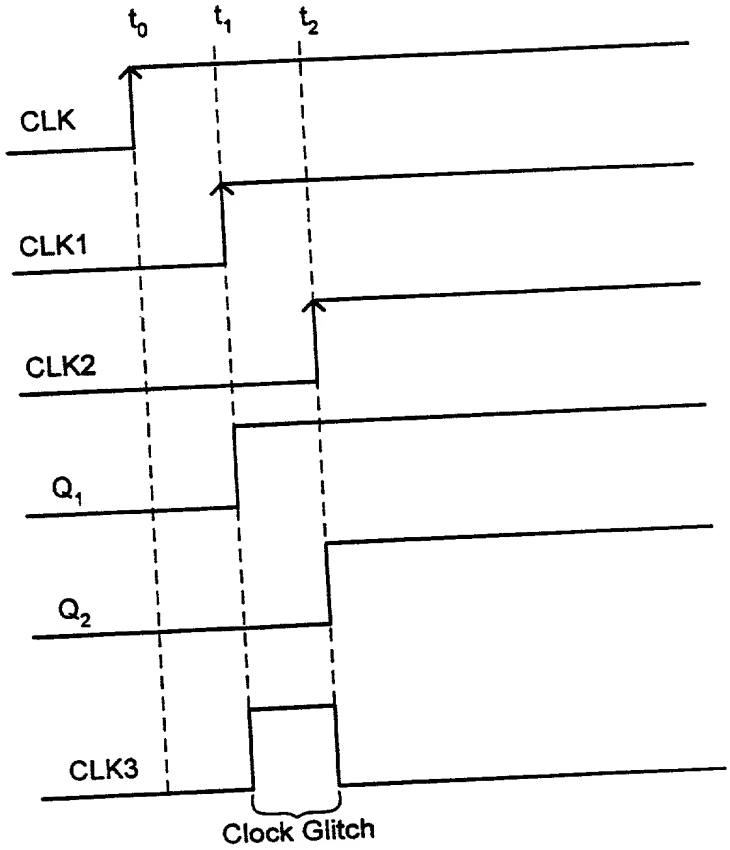
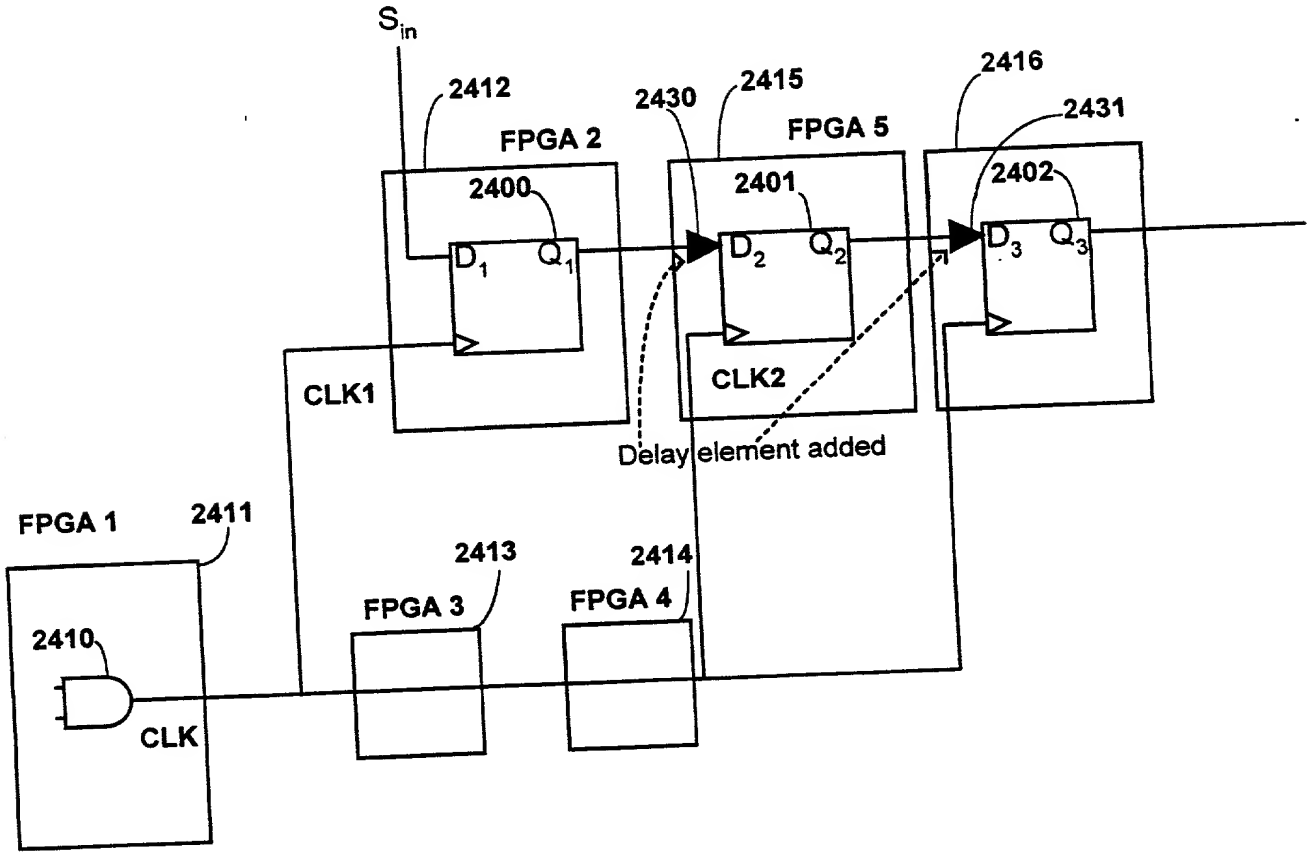


FIG. 77(B)

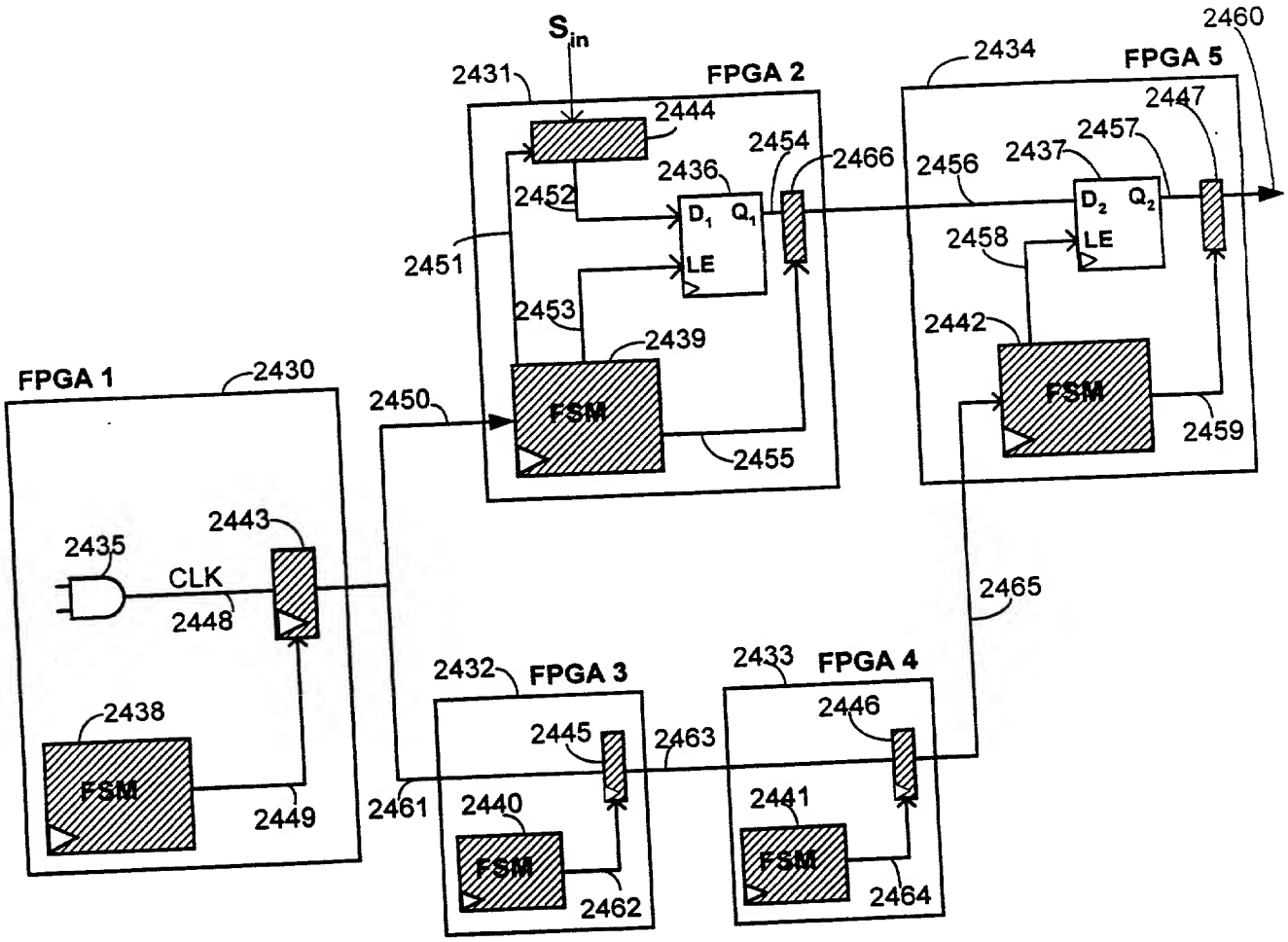
TIMING ADJUSTMENT BY ADDING DELAY



(Prior Art)

FIG. 78

GLOBAL RETIMING



Legend

Controlled by the global reference clock.

FSM and I/O registers for retiming control.

(Prior Art)

FIG. 79

TIGF LATCH

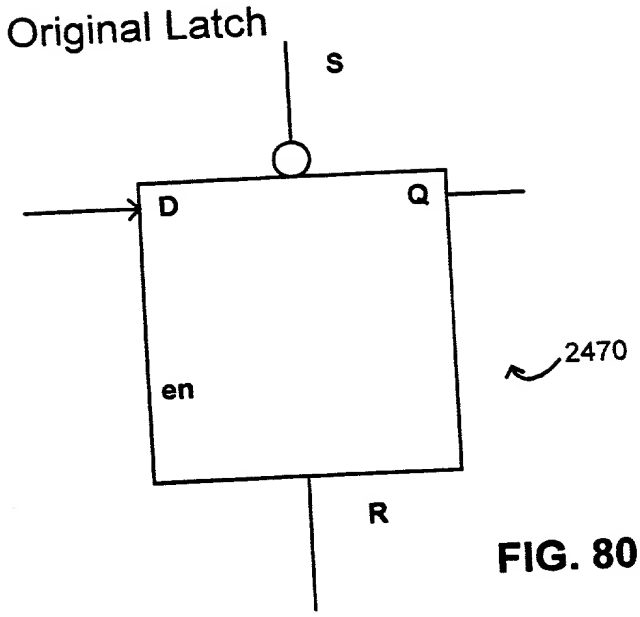


FIG. 80(A)

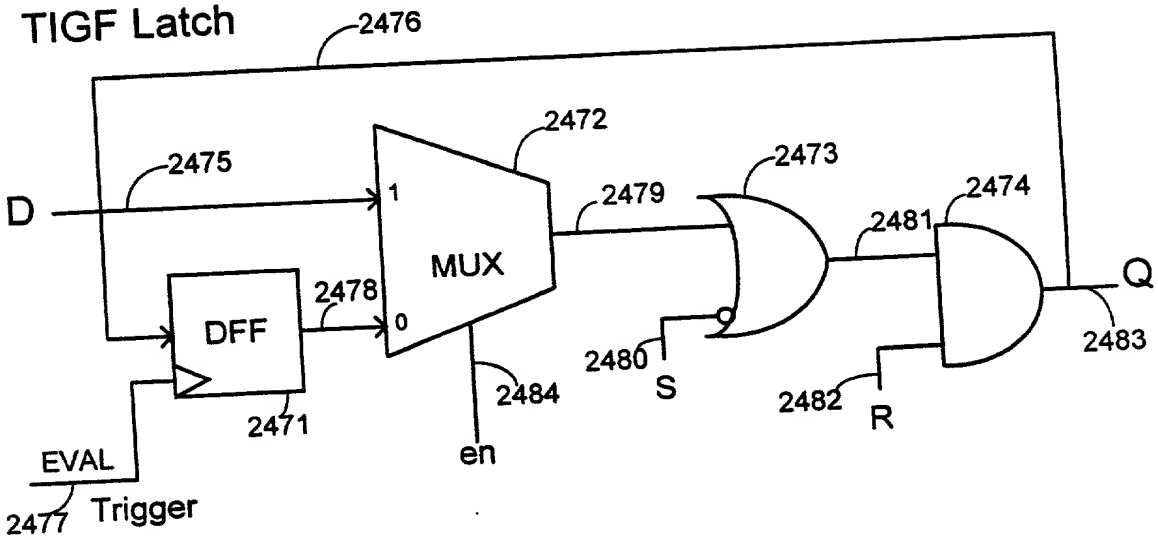


FIG. 80(B)

TIGF DFF

Original DFF

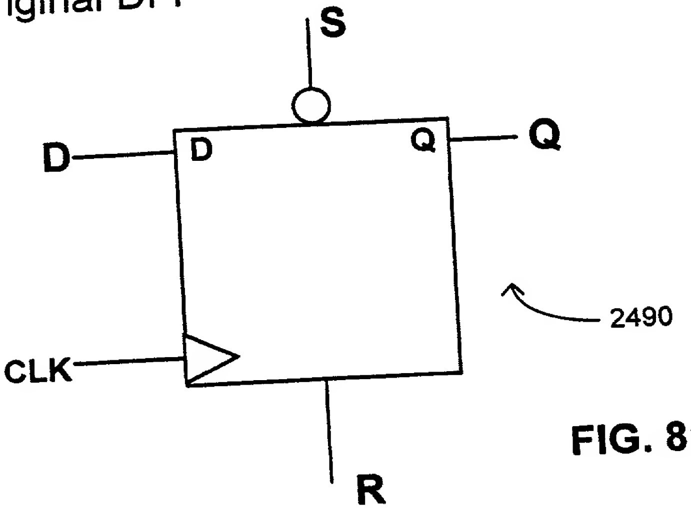


FIG. 81(A)

TIGF DFF and Edge Detector

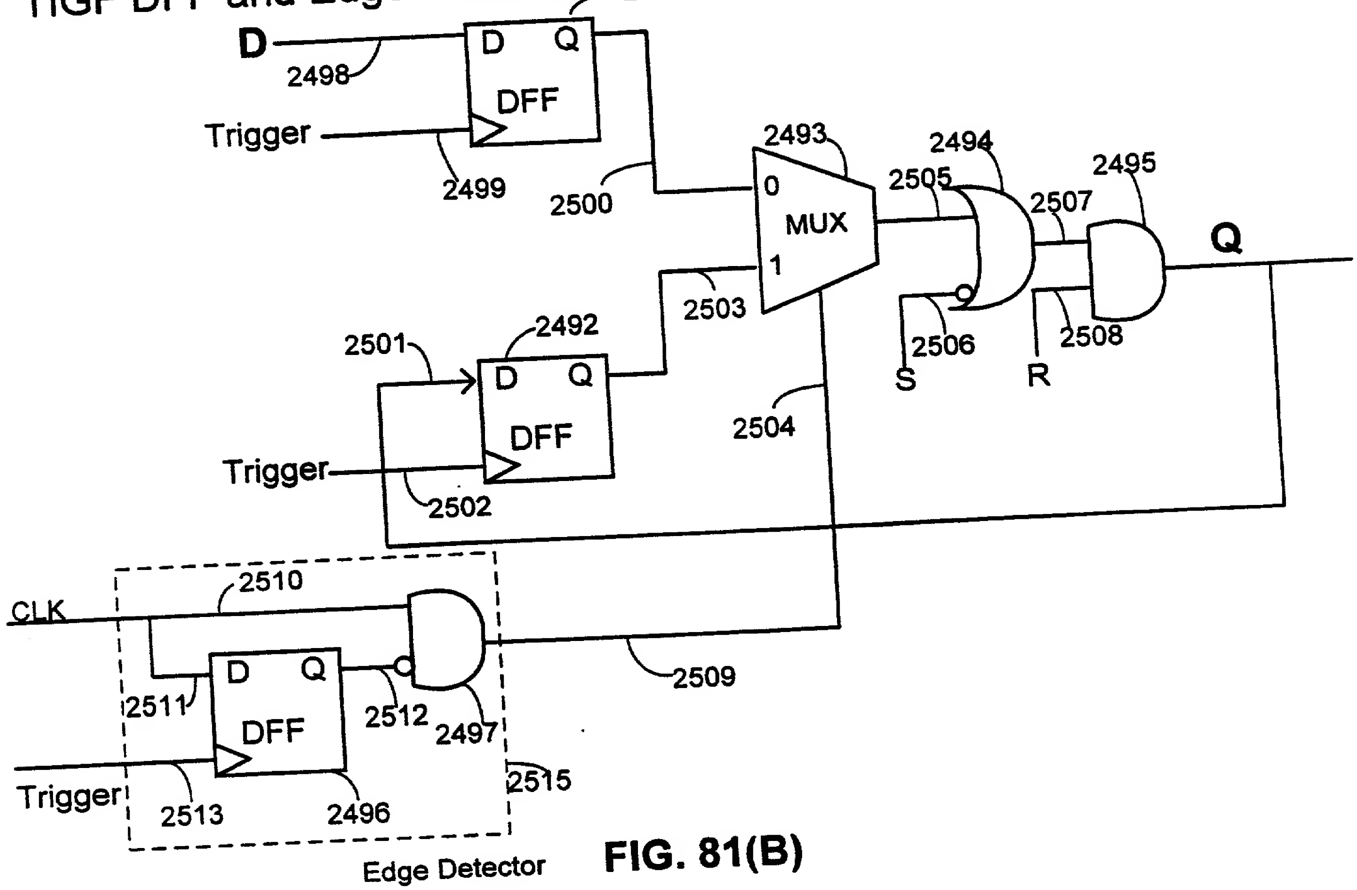


FIG. 81(B)

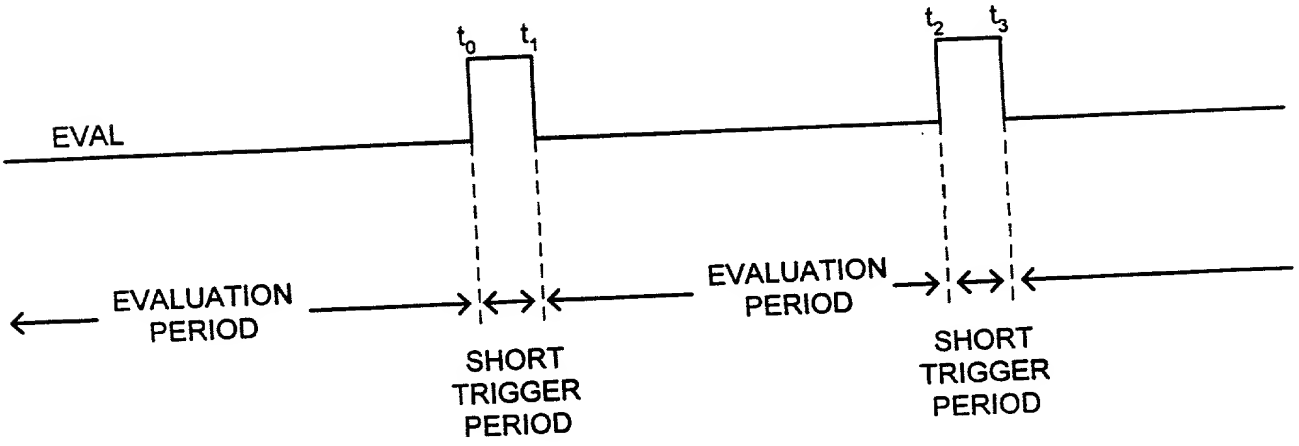

GLOBAL TRIGGER SIGNAL
EVAL
t_0
t_1
t_2
t_3
EVALUATION PERIOD
SHORT TRIGGER PERIOD
EVALUATION PERIOD
SHORT TRIGGER PERIOD

FIG. 82

RCC System

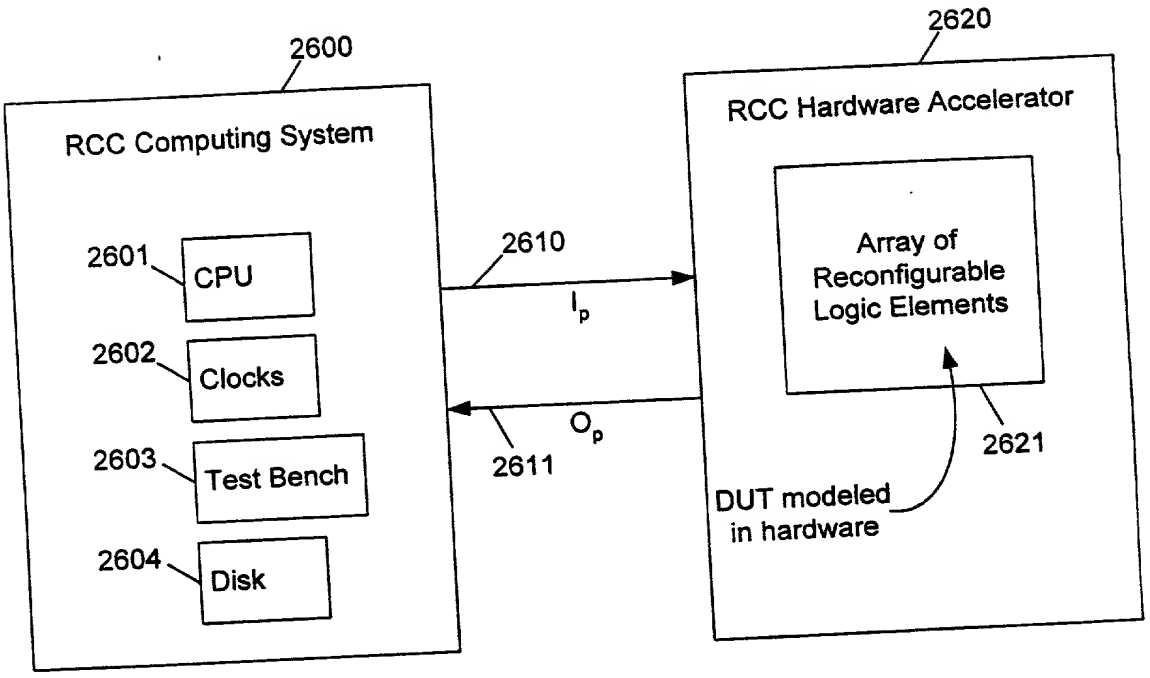


FIG. 83

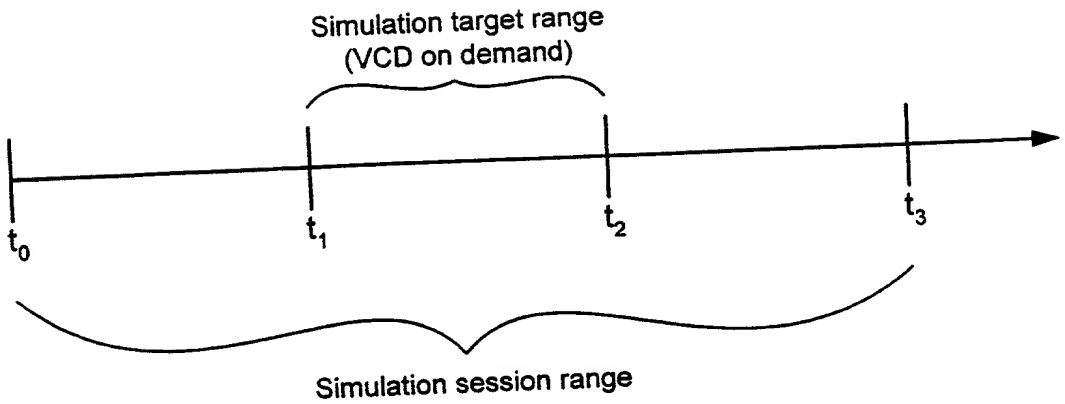


FIG. 84

SINGLE-ROW FPGA PER BOARD

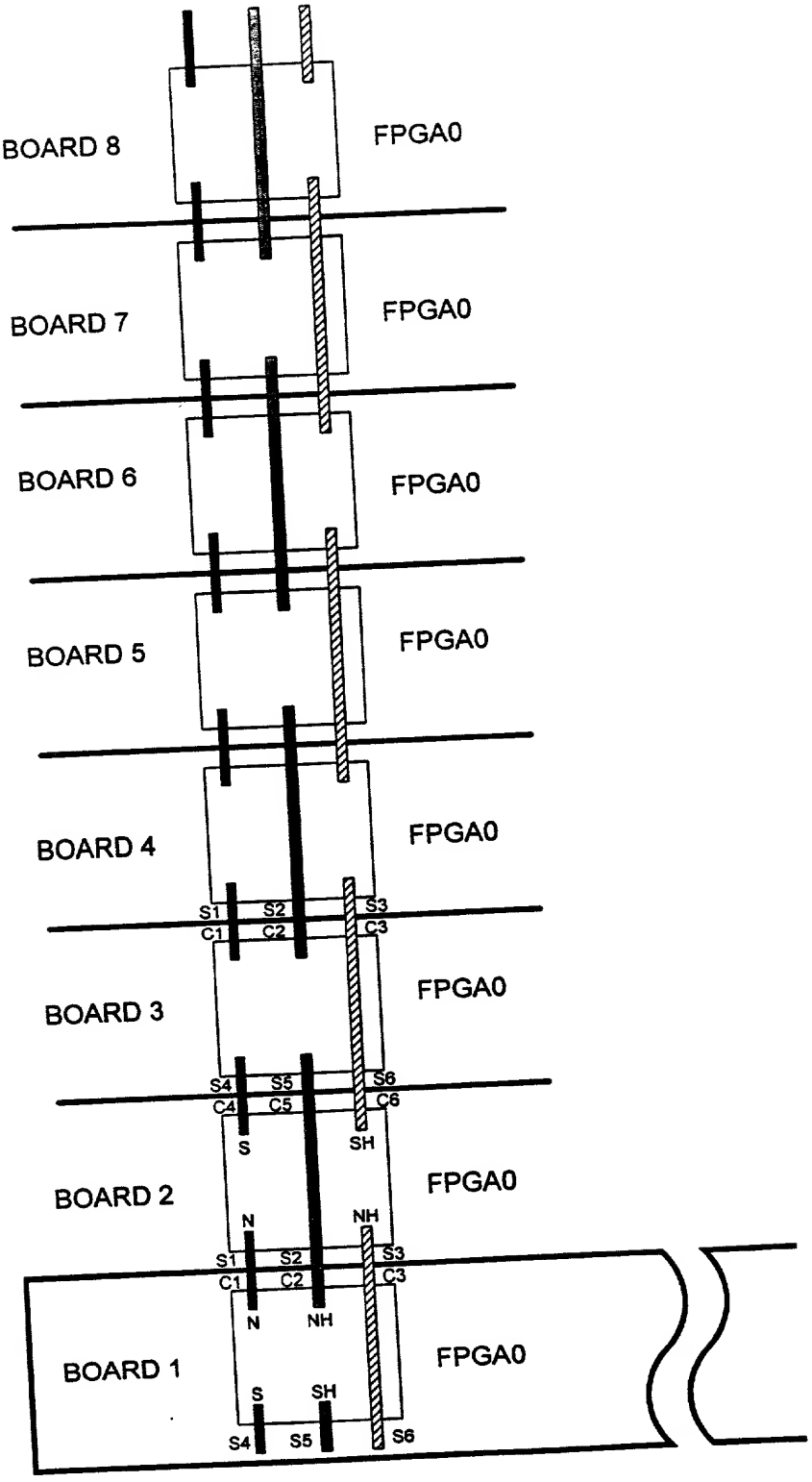


FIG. 85

TWO-ROW FPGA PER BOARD

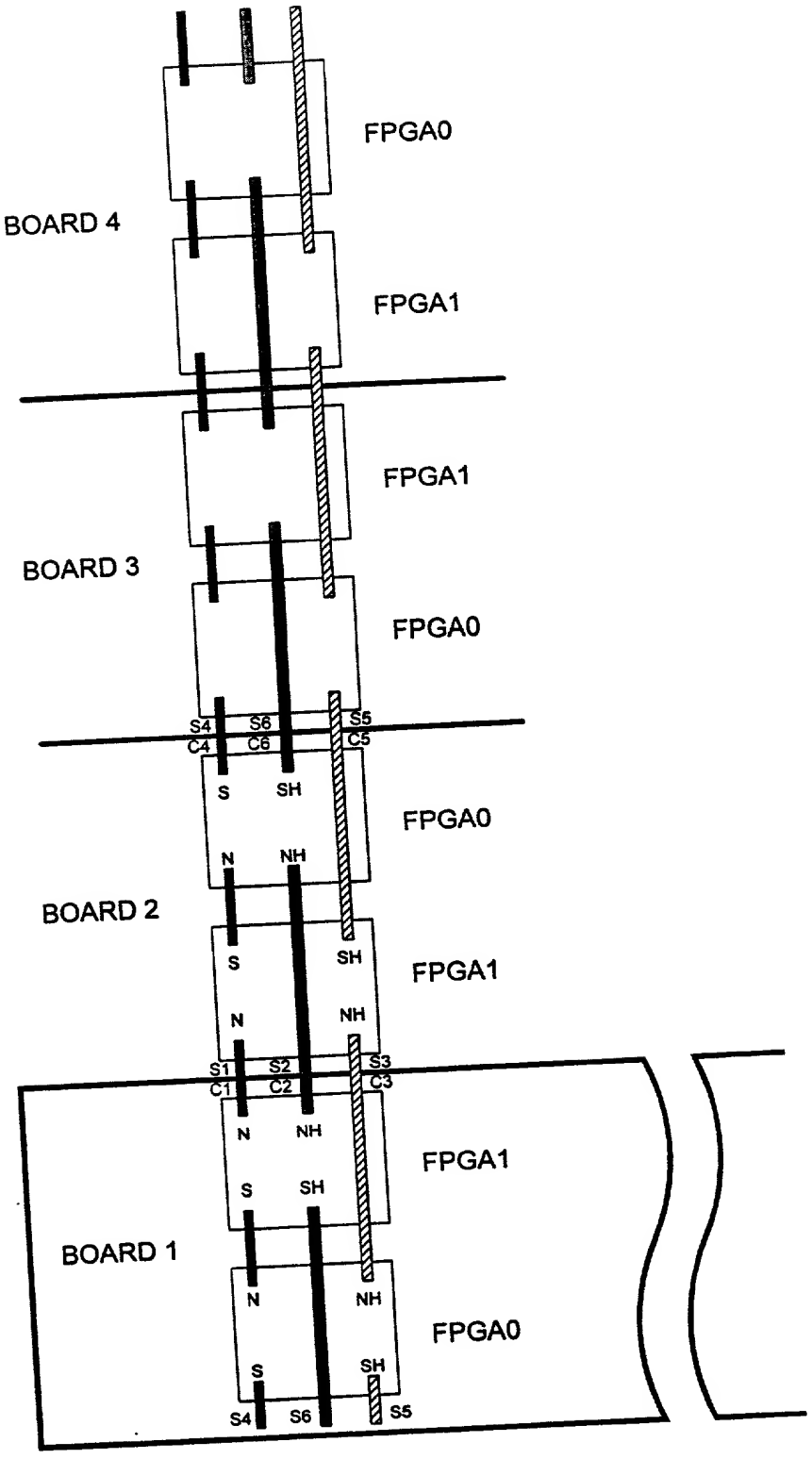


FIG. 86

THREE-ROW FPGA PER BOARD

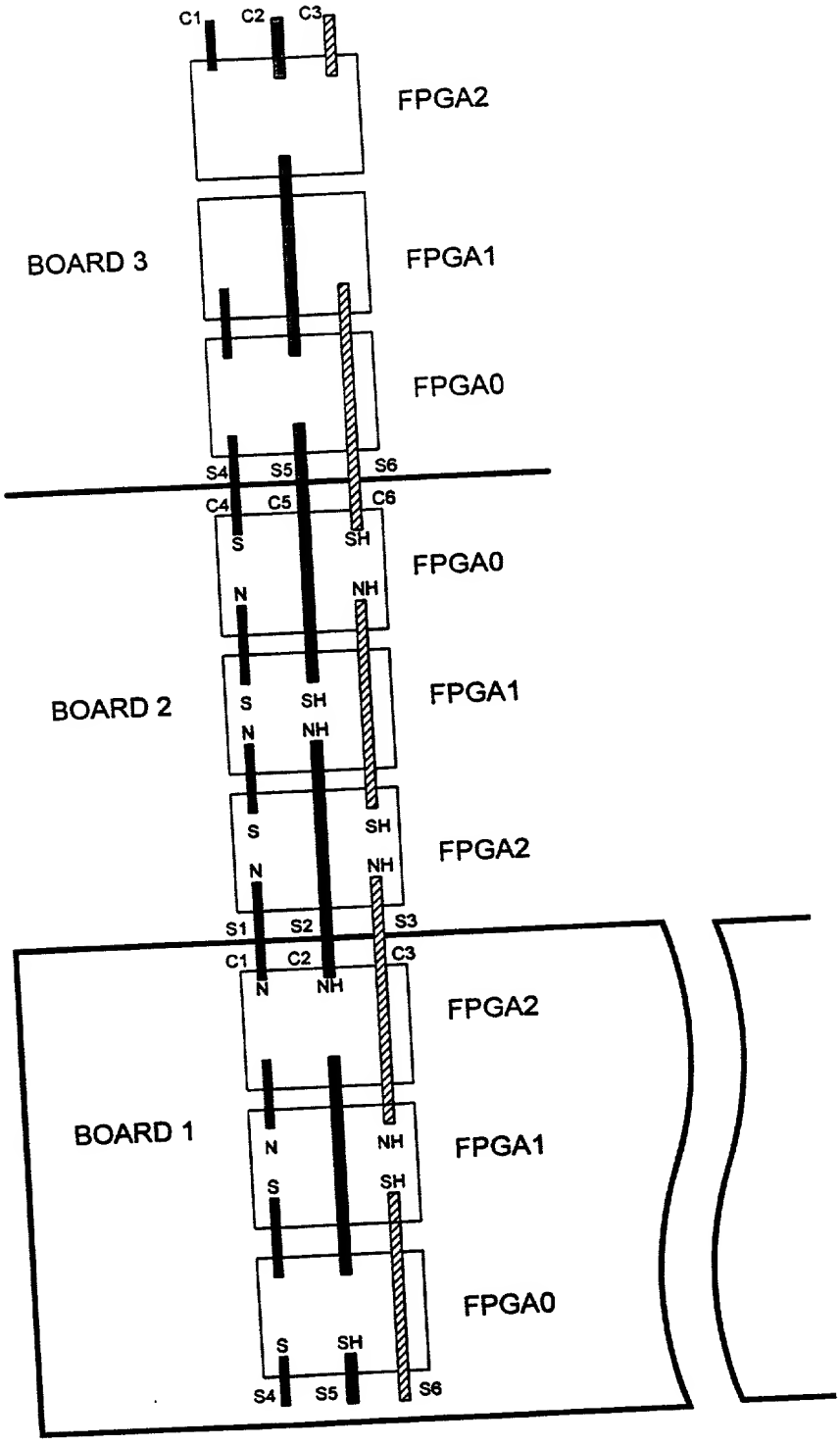


FIG. 87

FOUR-ROW FPGA PER BOARD

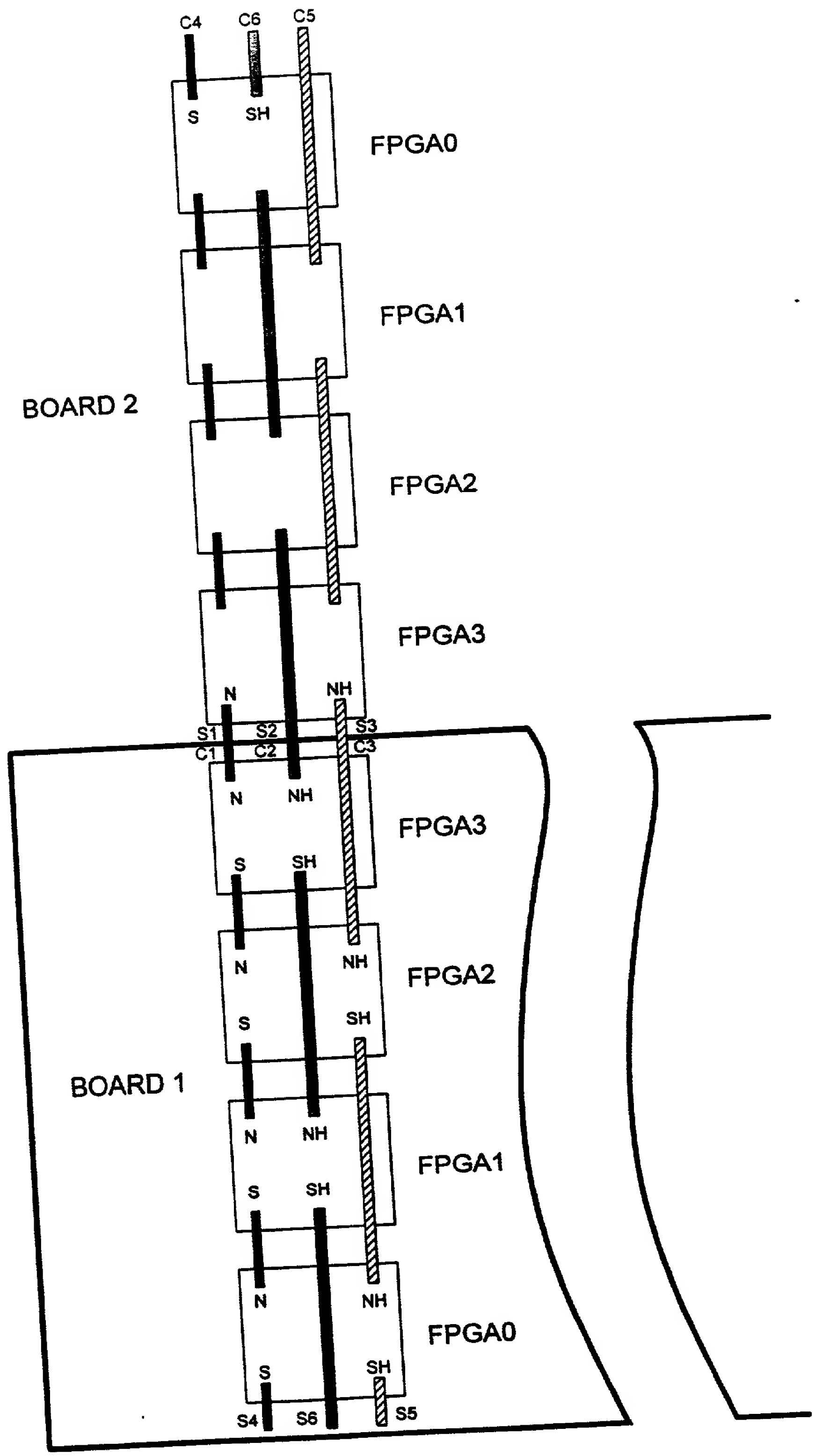


FIG. 88

INTERCONNECT FOR THREE-ROW PER BOARD

I/O Signals	Odd Board	Even Board	Common Board
	Connector-Group Pin-position	Connector-Group Pin-position	Connector-Group Pin-position
FPGA2_N	C1	S1	C1, S1
FPGA2_NH	C2	S3	C2, S3
FPGA1_NH	C3	S2	C3, S2
FPGA0_S	S4	C4	C4, S4
FPGA0_SH	S5	C6	C6, S5
FPGA1_SH	S6	C5	C5, S6

FIG. 89

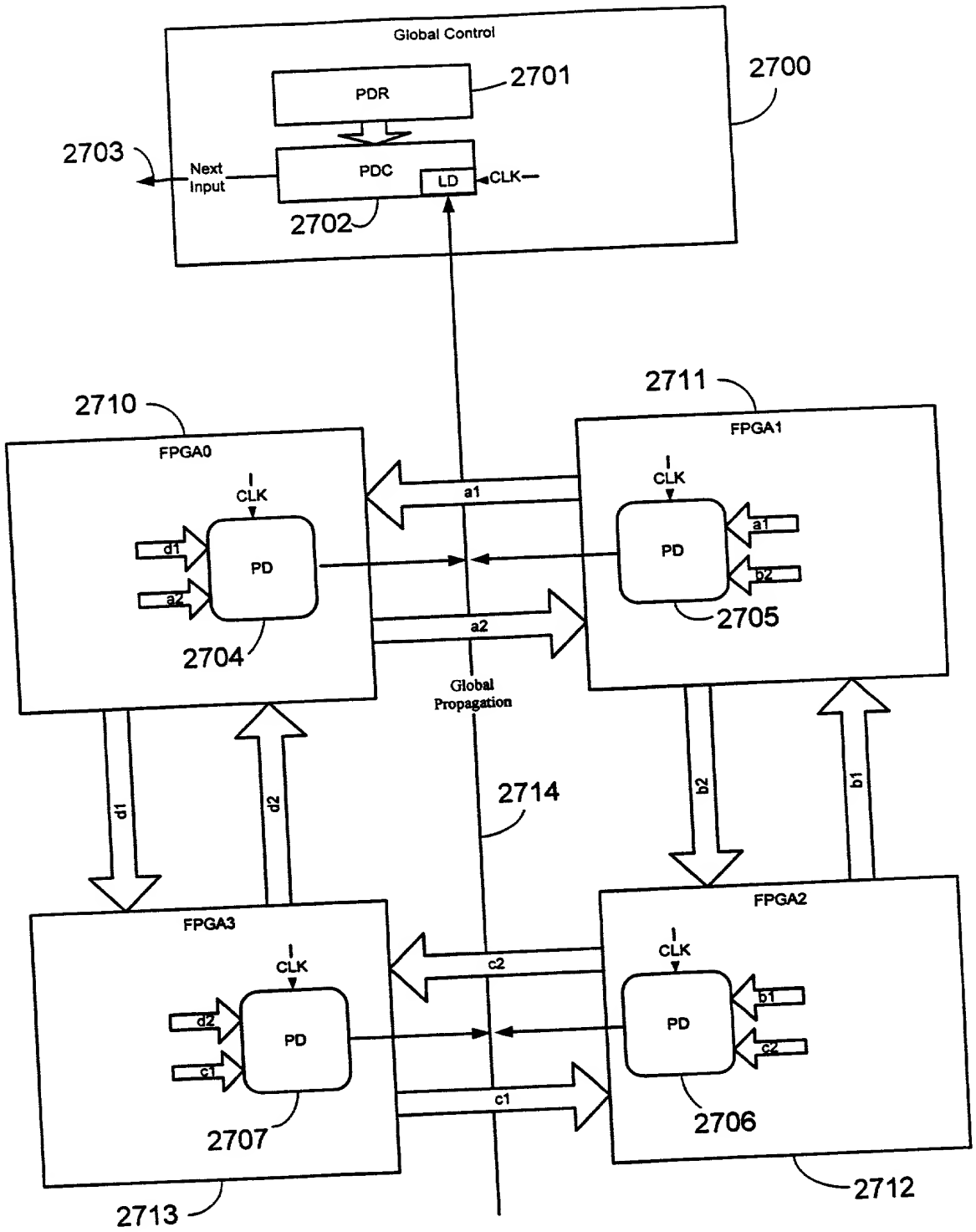

Global Control
PDR
2701
2700
2703
Next Input
PDC
LD
CLK
2702
2710
FPGA0
CLK
d1
PD
a2
2704
2711
FPGA1
CLK
a1
PD
b2
2705
a1
a2
Global Propagation
2714
d1
d2
b2
b1
FPGA3
CLK
d2
PD
c1
2707
2713
FPGA2
CLK
c2
b1
PD
c2
2706
c1
2712

FIG. 90

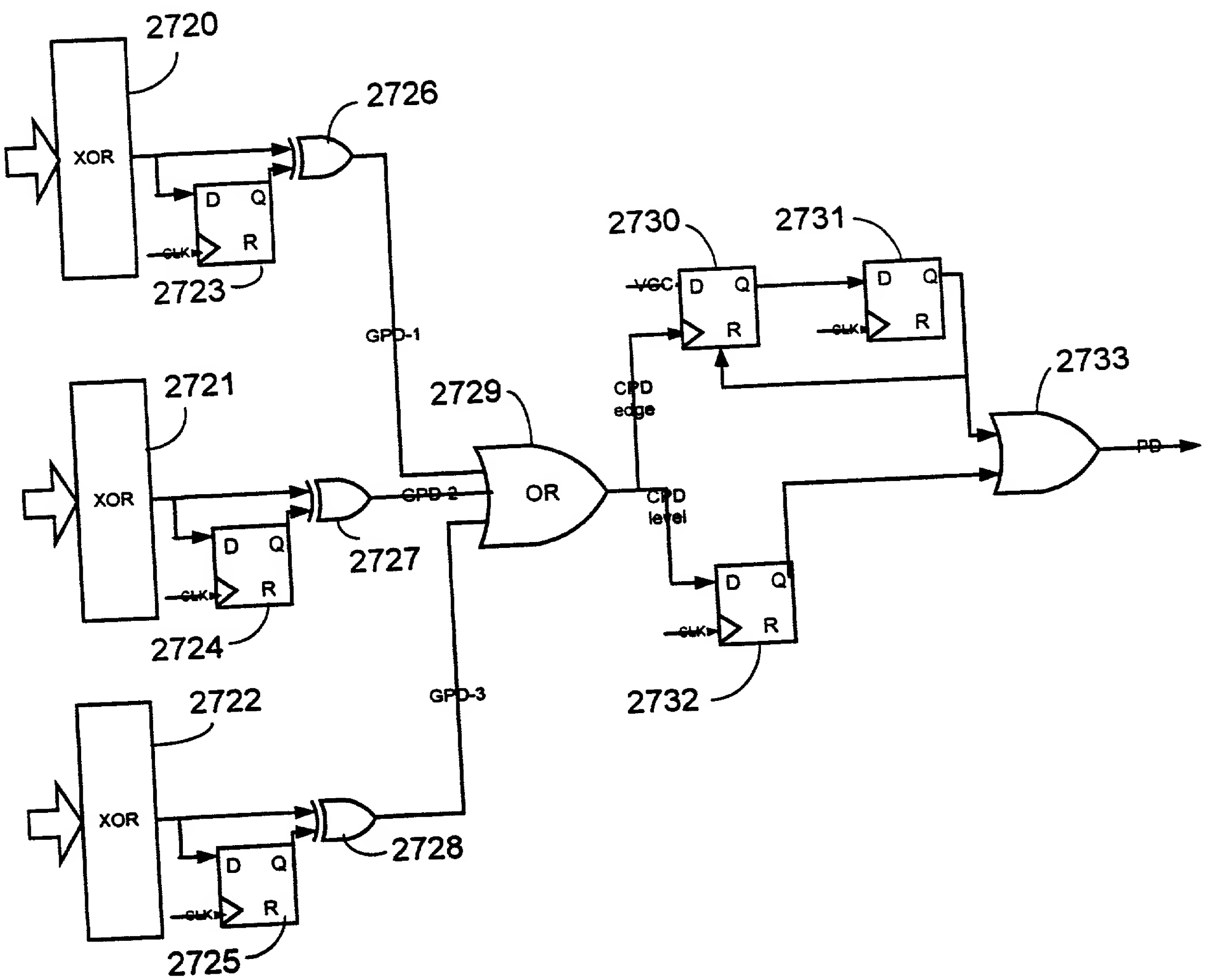

2720
XOR
2726
2723
CLK
D
Q
R
GPD-1
2721
XOR
2727
2724
GPD-2
GPD-3
2722
XOR
2728
2725
2729
OR
2730
2731
VCC
CPD
edge
CPD
level
2732
2733
PD

FIG. 91

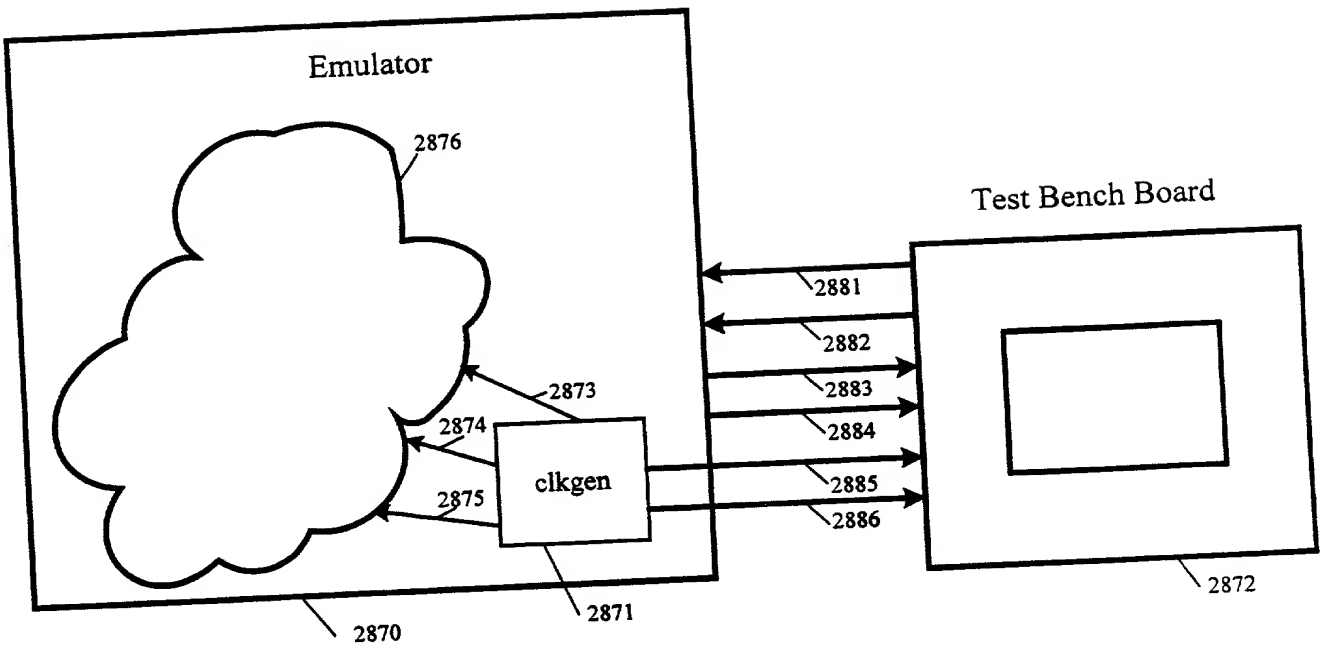

Emulator
2876
Test Bench Board
2881
2882
2883
2884
2885
2886
2873
2874
clkgen
2875
2871
2870
2872

FIG. 92

Clock Specification

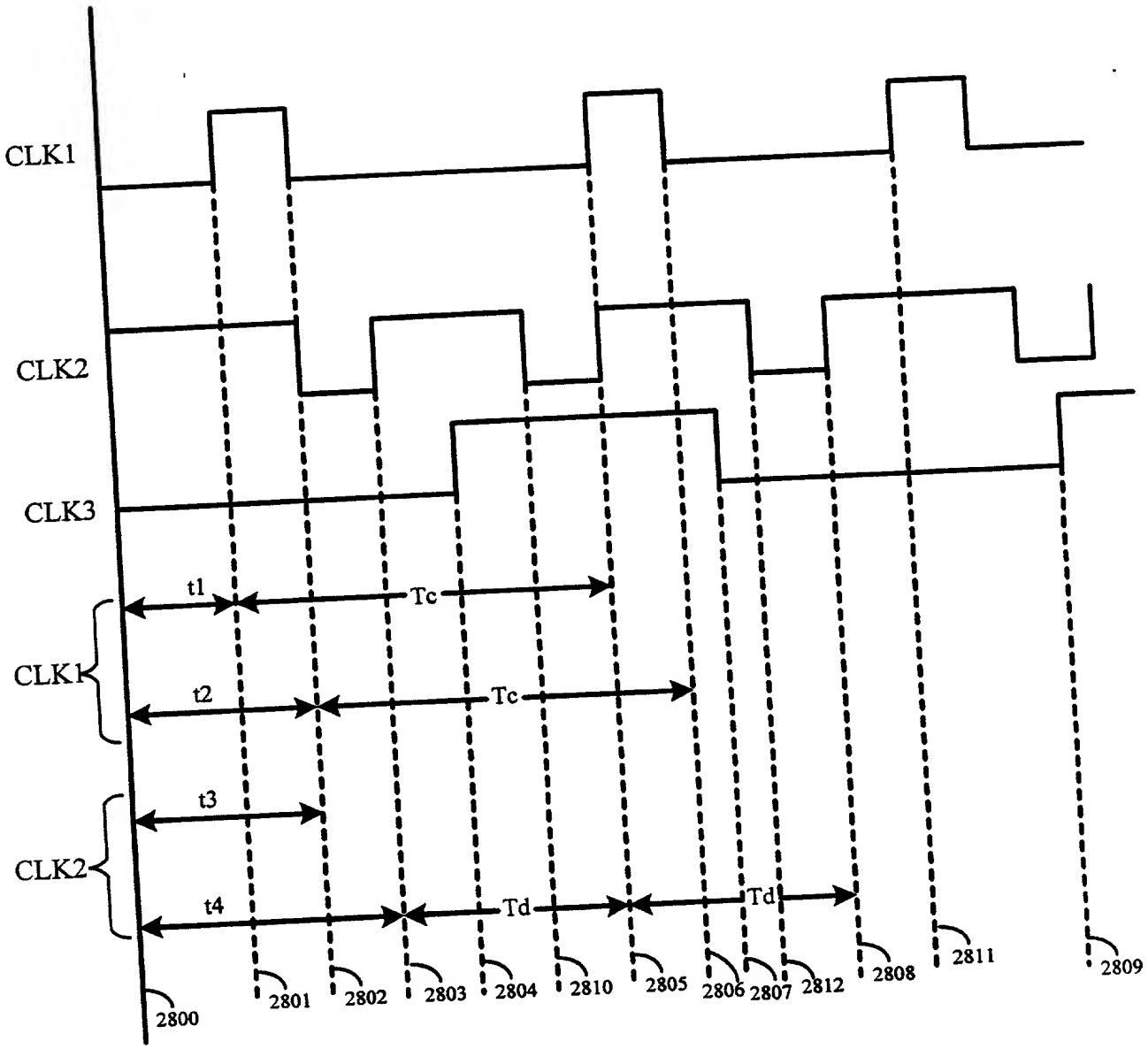


FIG. 93

Clock Generation Scheduler w/ Slices

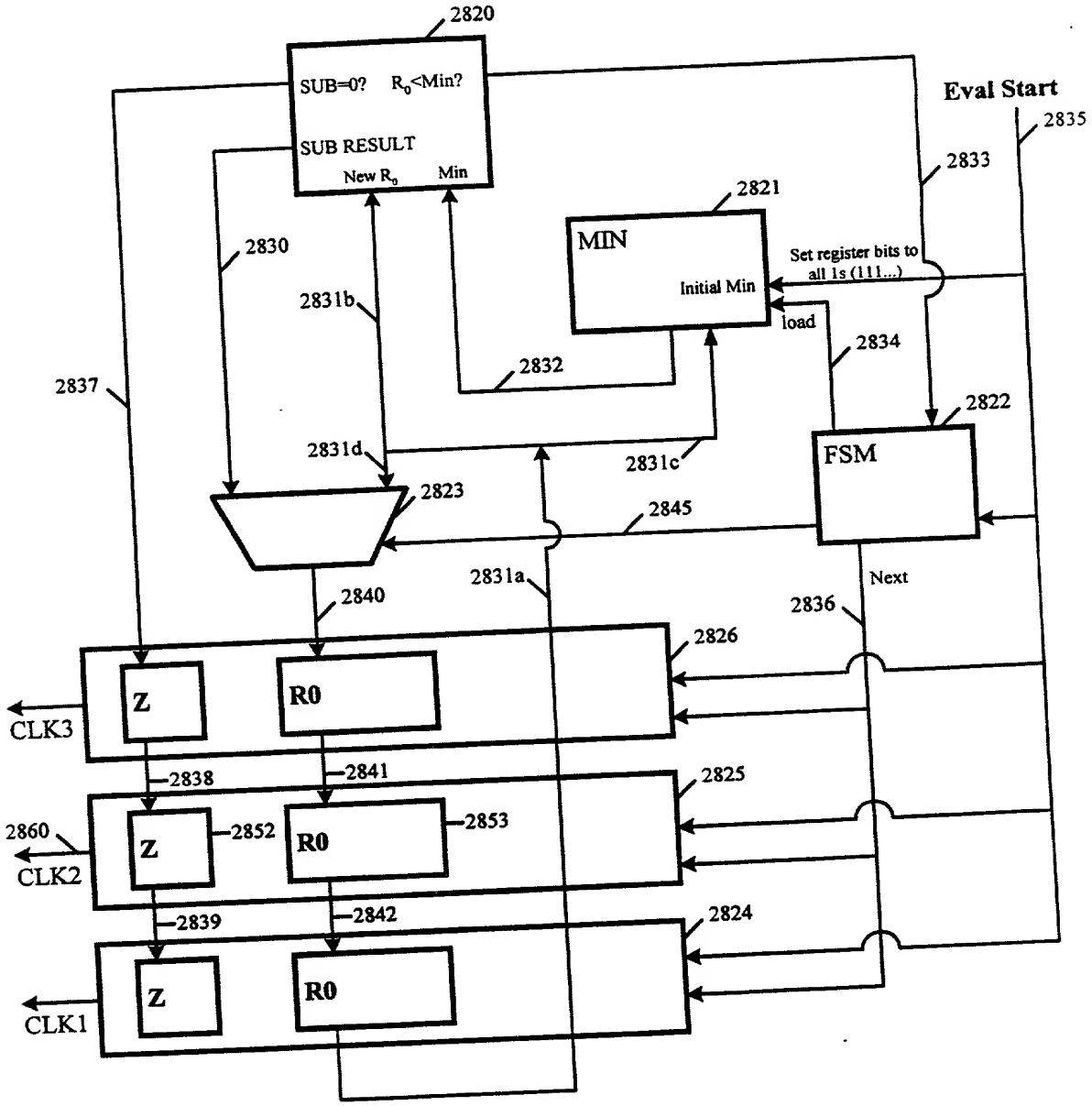


FIG. 94

Clock Generation Slice

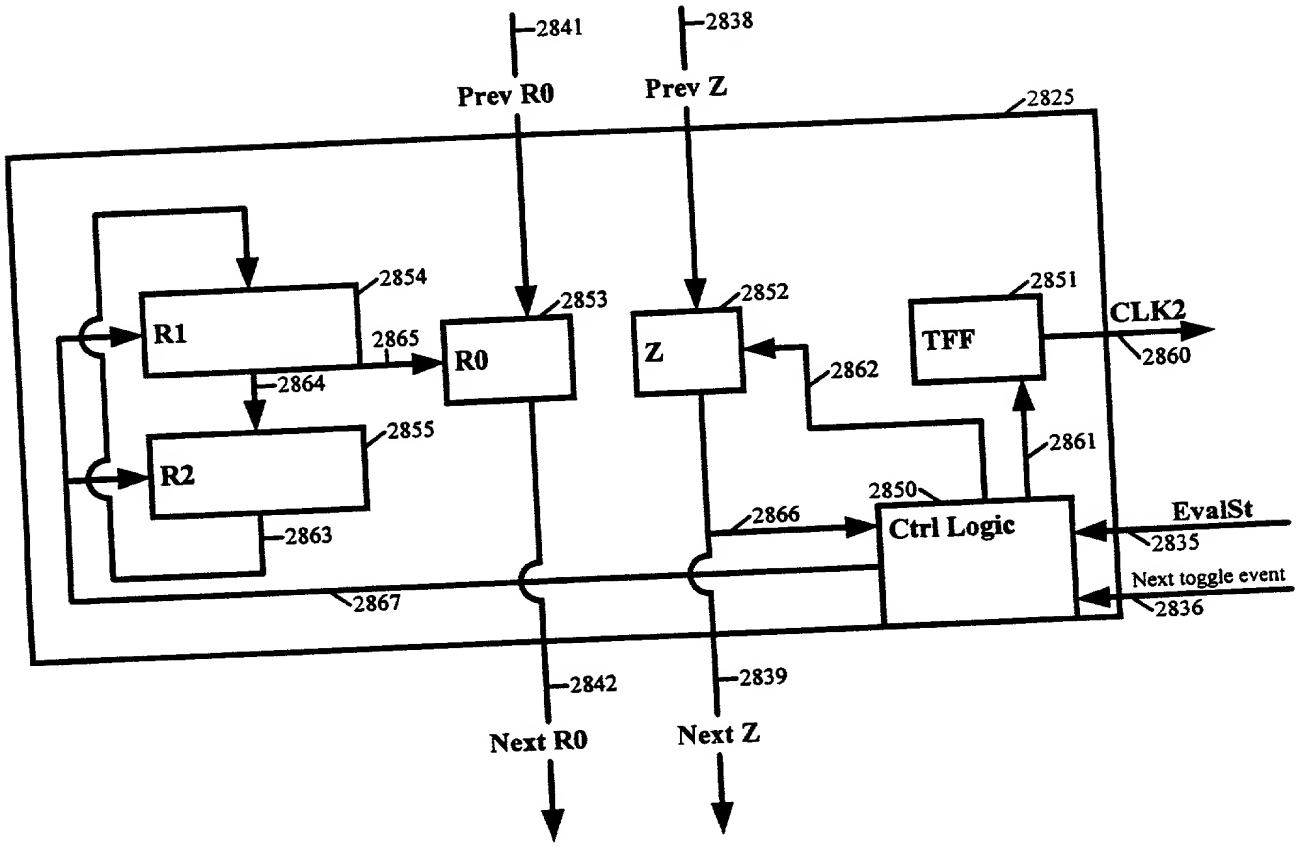


FIG. 95

Clock Generation Scheduler and Slices

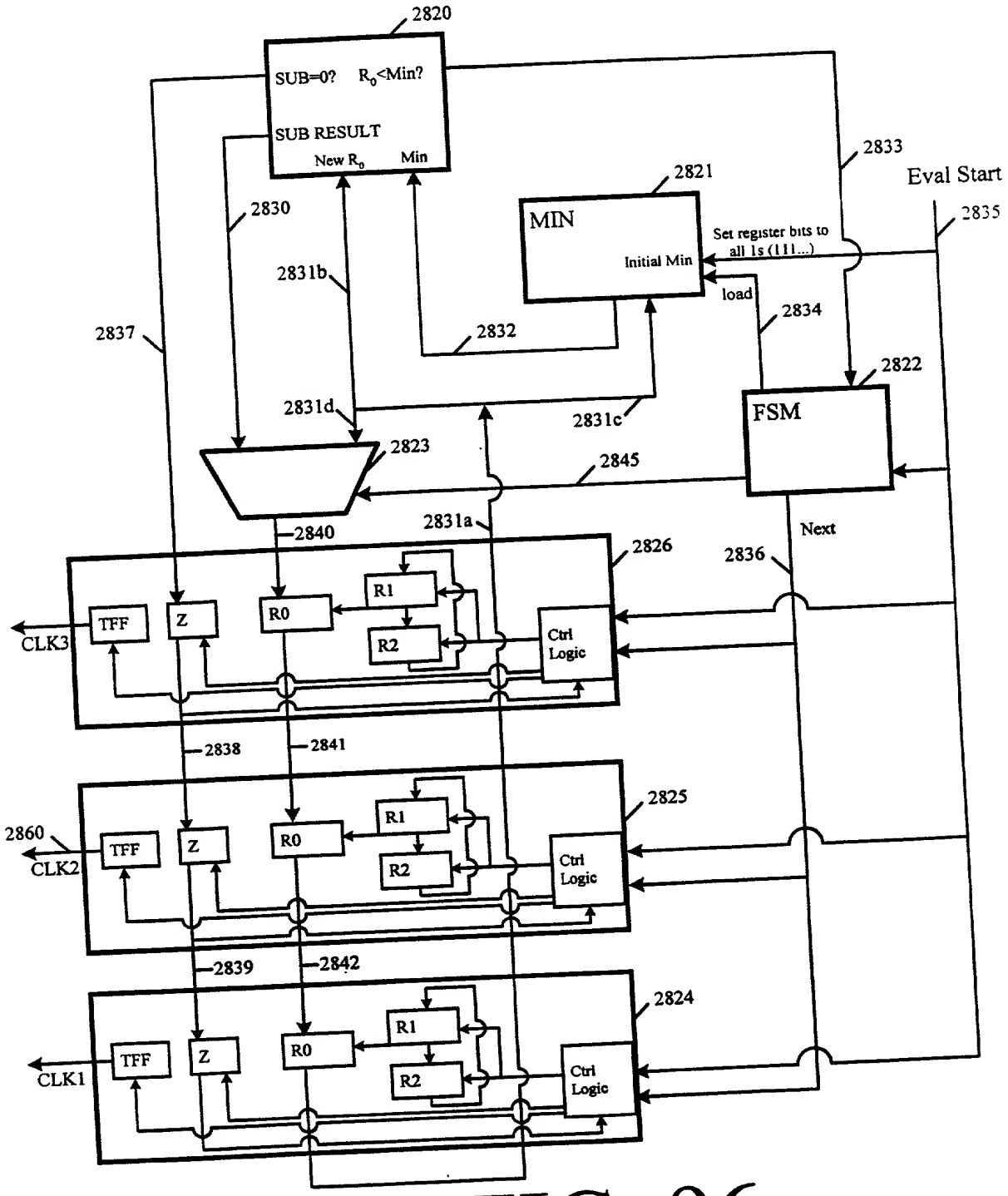


FIG. 96

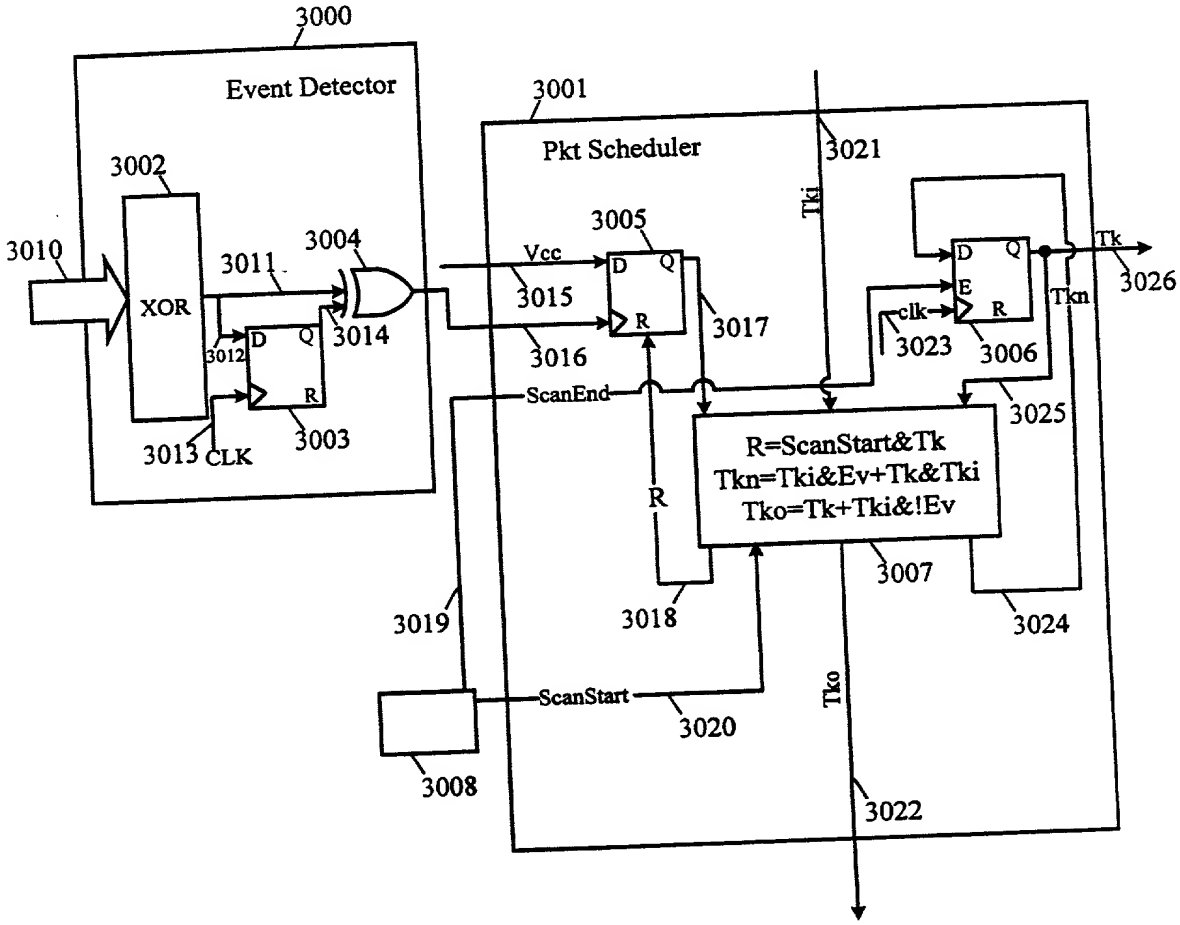

3000
Event Detector
3001
Pkt Scheduler
3002
3010
XOR
3011
3004
3012
D
Q
3014
R
3003
3013
CLK
3005
Vcc
3015
3016
3017
3021
Tki
clk
3023
3006
E
Tkn
Tk
3026
ScanEnd
3025
R=ScanStart&Tk
Tkn=Tki&Ev+Tk&Tki
Tko=Tk+Tki&!Ev
R
3018
3007
3024
3019
ScanStart
3020
Tko
3008
3022

FIG. 97

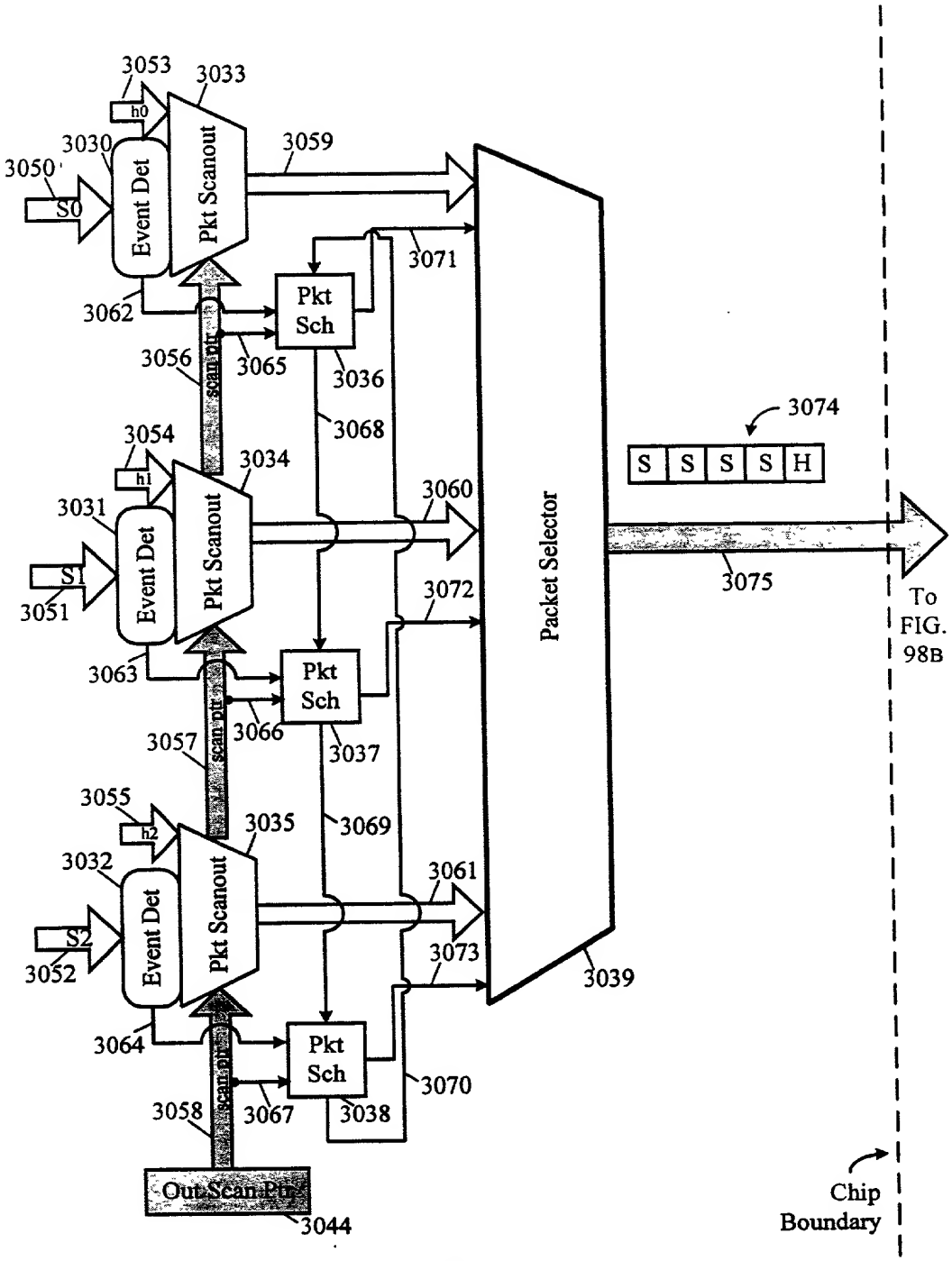

3053
3033
h0
3030
3050
S0
Event Det
Pkt Scanout
3059
3071
Pkt Sch
3062
3056
scan out
3065
3036
3068
3054
3034
h1
3031
3060
S1
3051
Event Det
Pkt Scanout
3072
3063
Pkt Sch
3066
3057
scan out
3037
3055
3035
3069
h2
3032
3061
S2
3052
Event Det
Pkt Scanout
3073
3039
3064
Pkt Sch
3070
3058
scan out
3067
3038
Out Scan Pin
3044
Packet Selector
3074
S S S S H
3075
To FIG. 98B
Chip Boundary

FIG. 98A

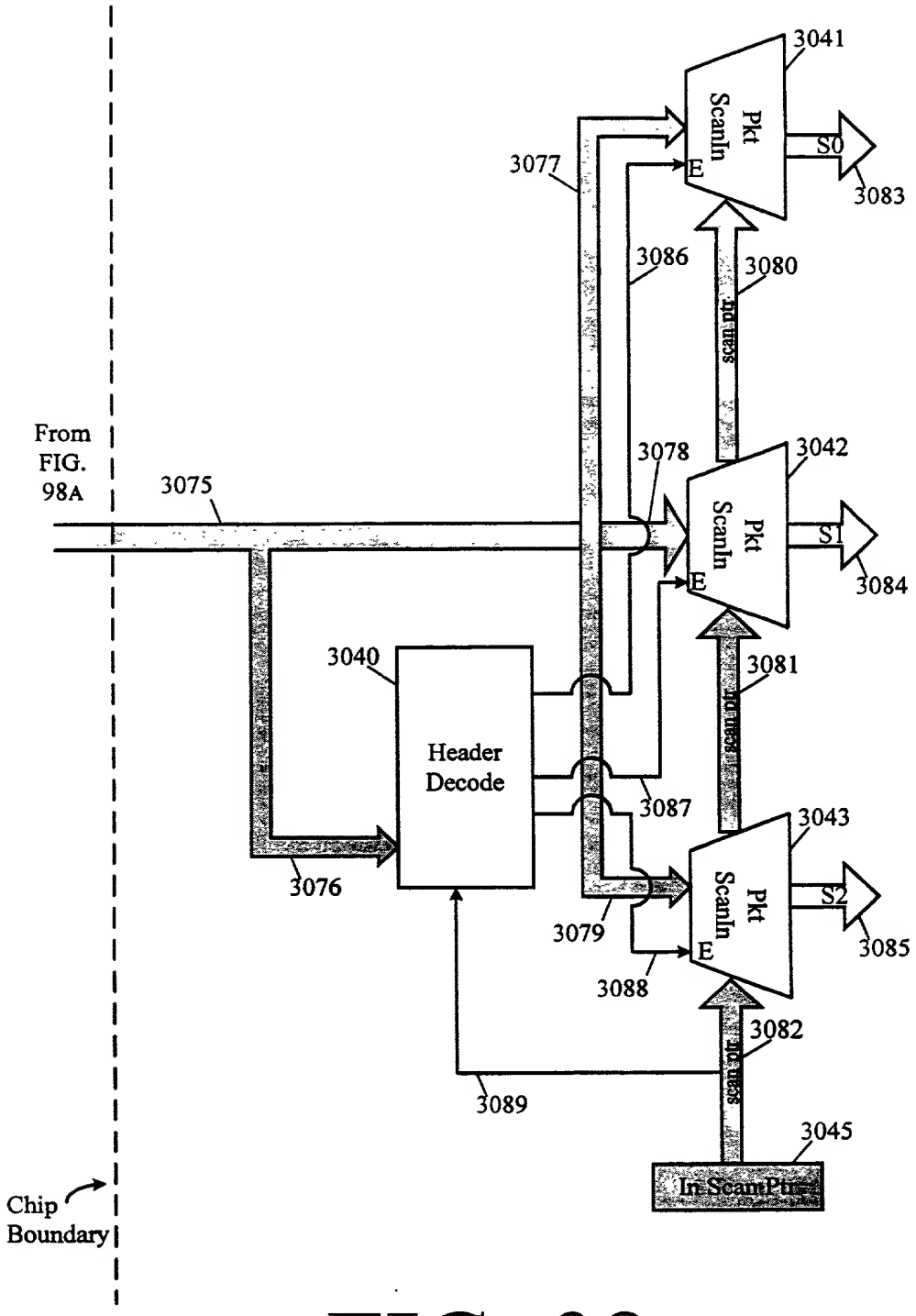

From FIG. 98A
3075
3076
3077
3078
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3040
Header Decode
3041
3042
3043
3045
Pkt ScanIn
S0
S1
S2
E
scan out
In ScanPus
Chip Boundary

FIG. 98B

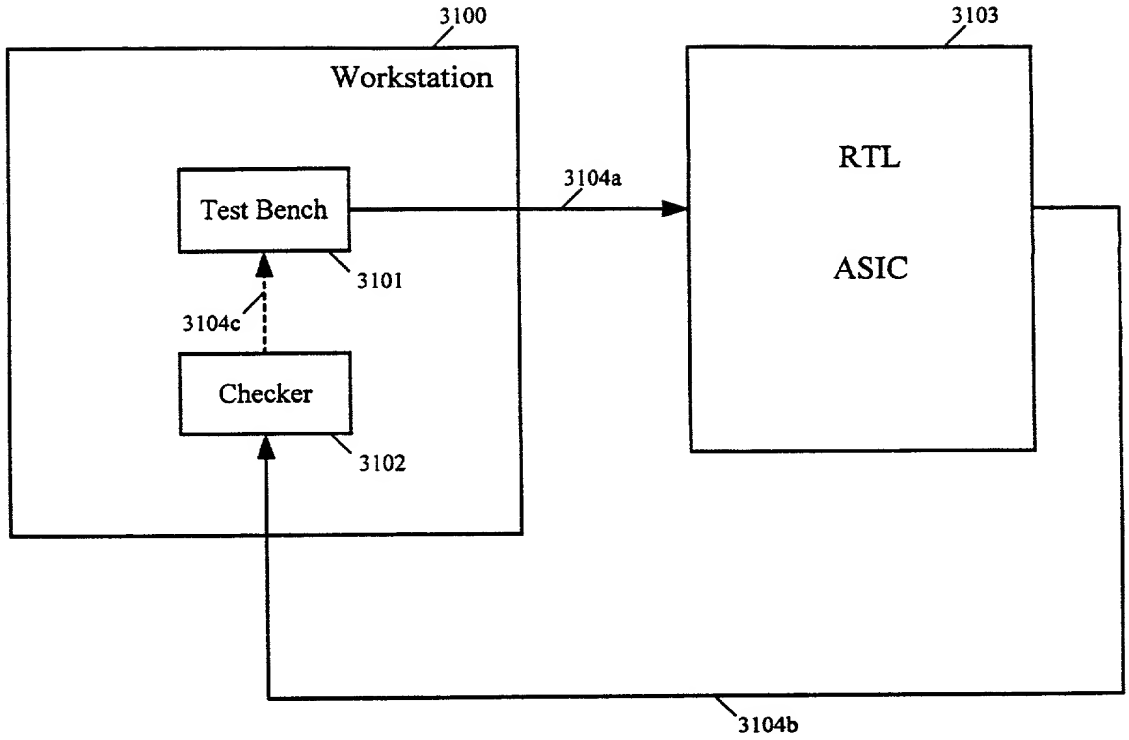

3100
Workstation
Test Bench
3101
3104c
Checker
3102
3104a
3103
RTL
ASIC
3104b

FIG. 99

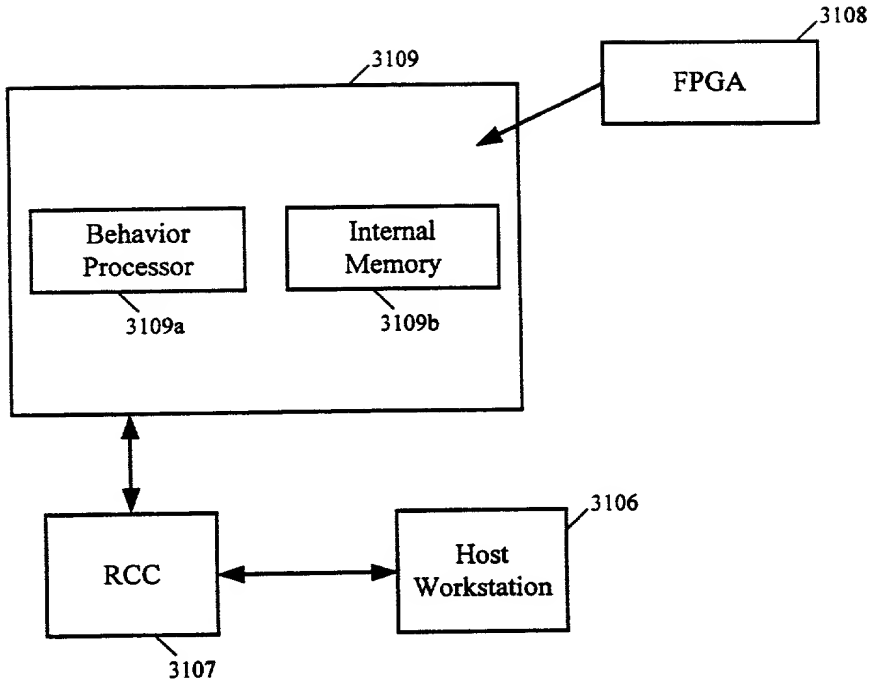

3108
FPGA
3109
Behavior Processor
3109a
Internal Memory
3109b
3106
Host Workstation
RCC
3107

FIG. 100

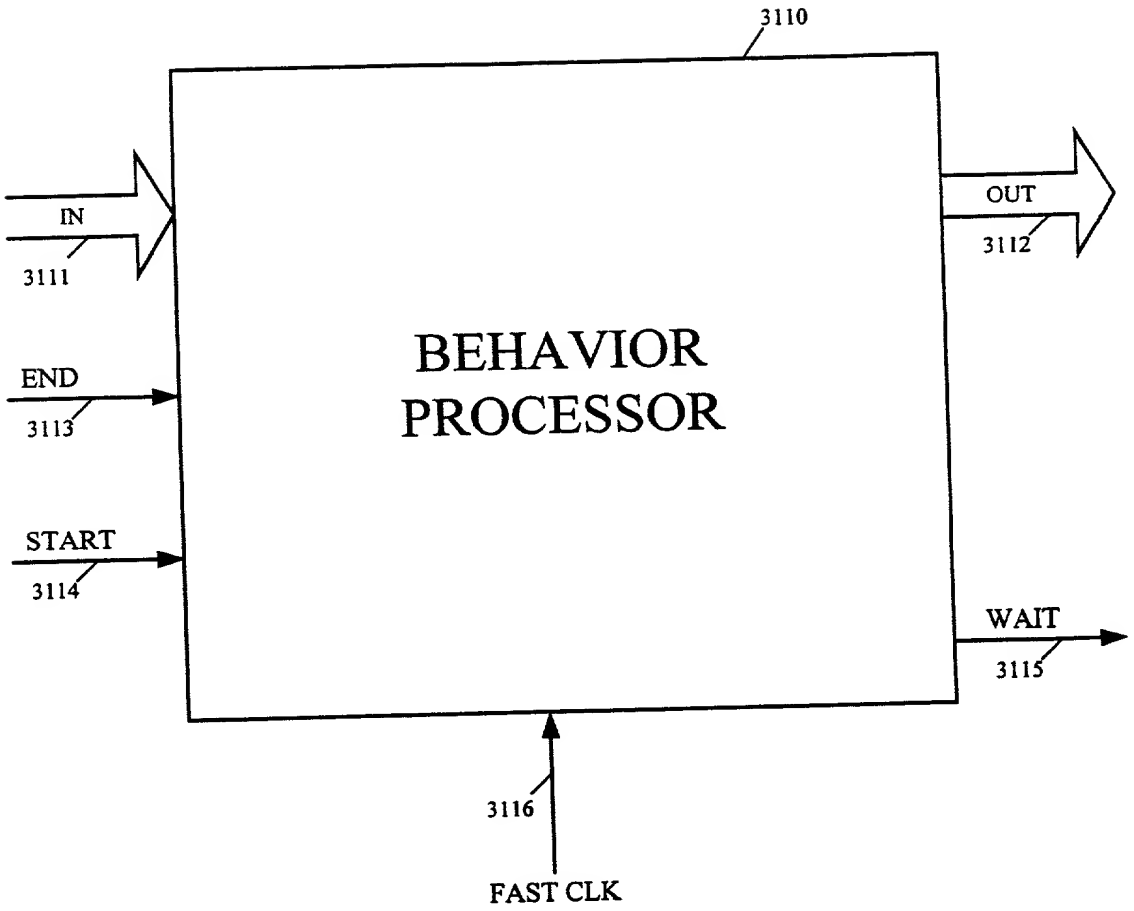

3110
BEHAVIOR
PROCESSOR
IN
3111
END
3113
START
3114
OUT
3112
WAIT
3115
3116
FAST CLK

FIG. 101

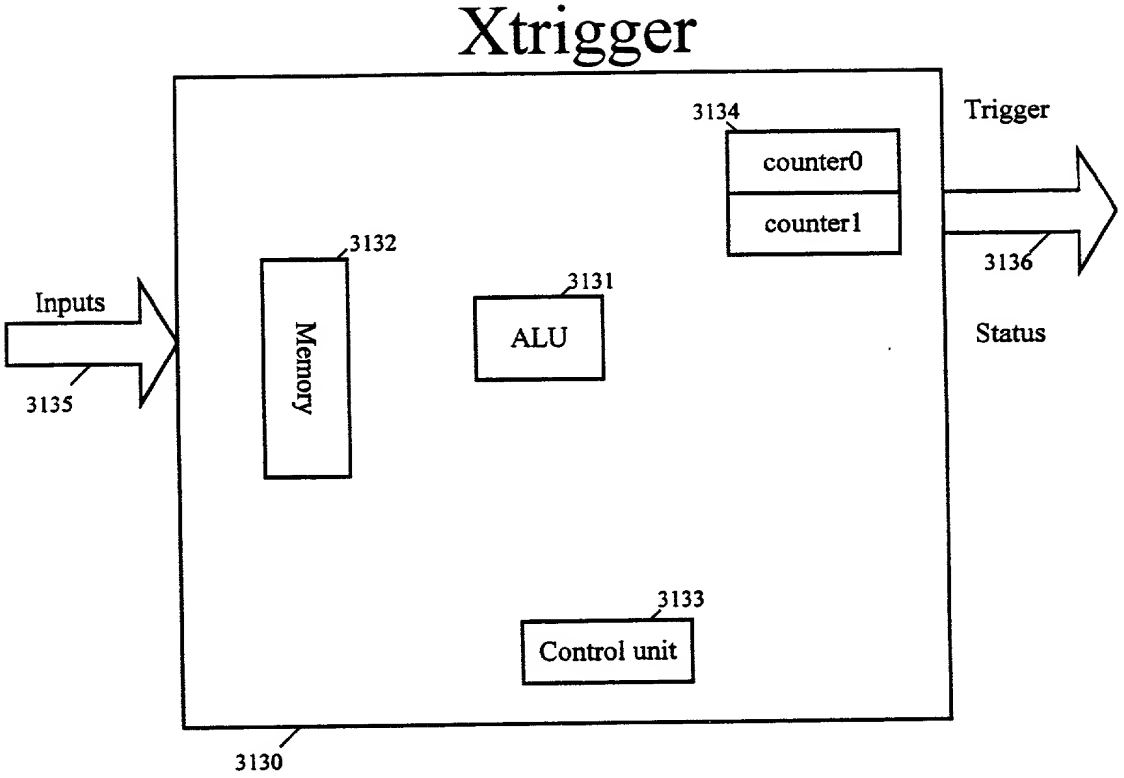

Xtrigger
3134
counter0
counter1
Trigger
3136
Status
3132
Memory
3131
ALU
Inputs
3135
3133
Control unit
3130

FIG. 105

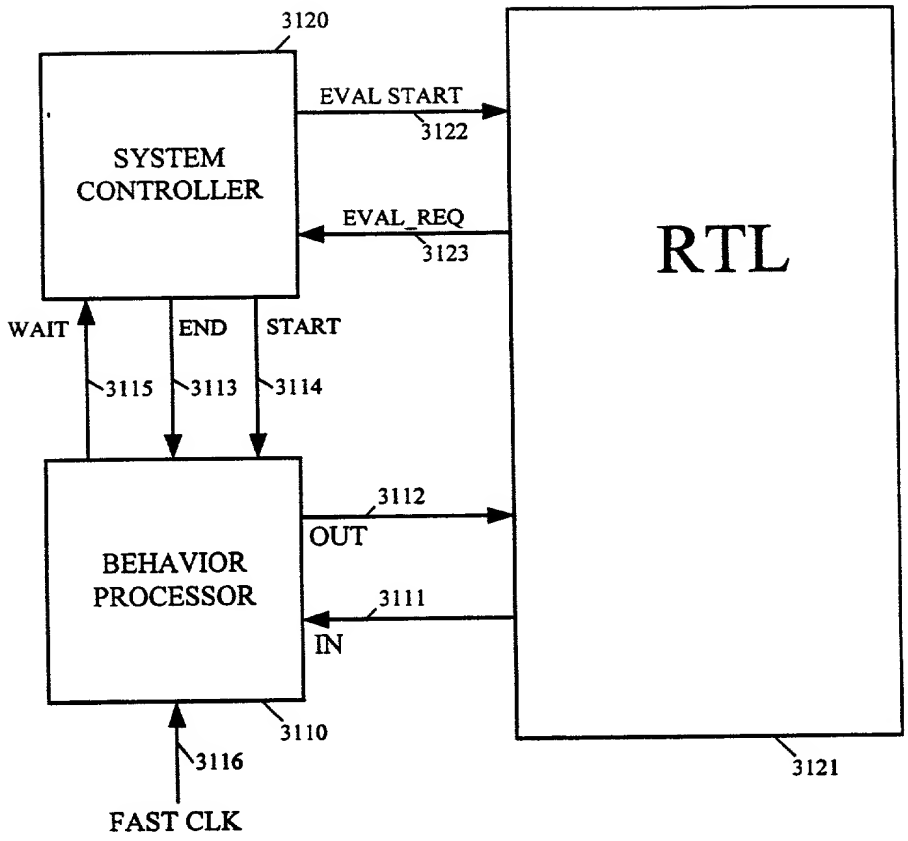

3120
SYSTEM CONTROLLER
EVAL START
3122
EVAL_REQ
3123
RTL
WAIT
END
START
3115
3113
3114
BEHAVIOR PROCESSOR
3112
OUT
3111
IN
3110
3116
FAST CLK
3121

FIG. 102

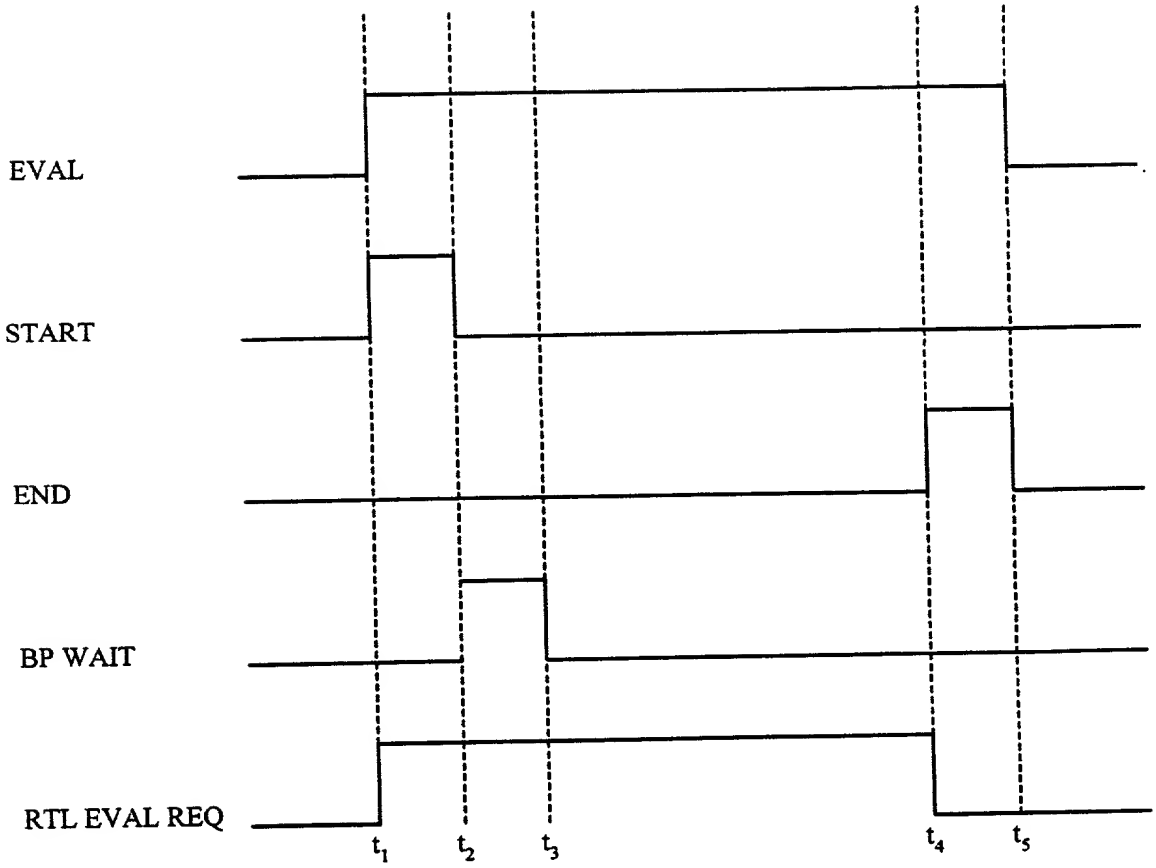


FIG. 103

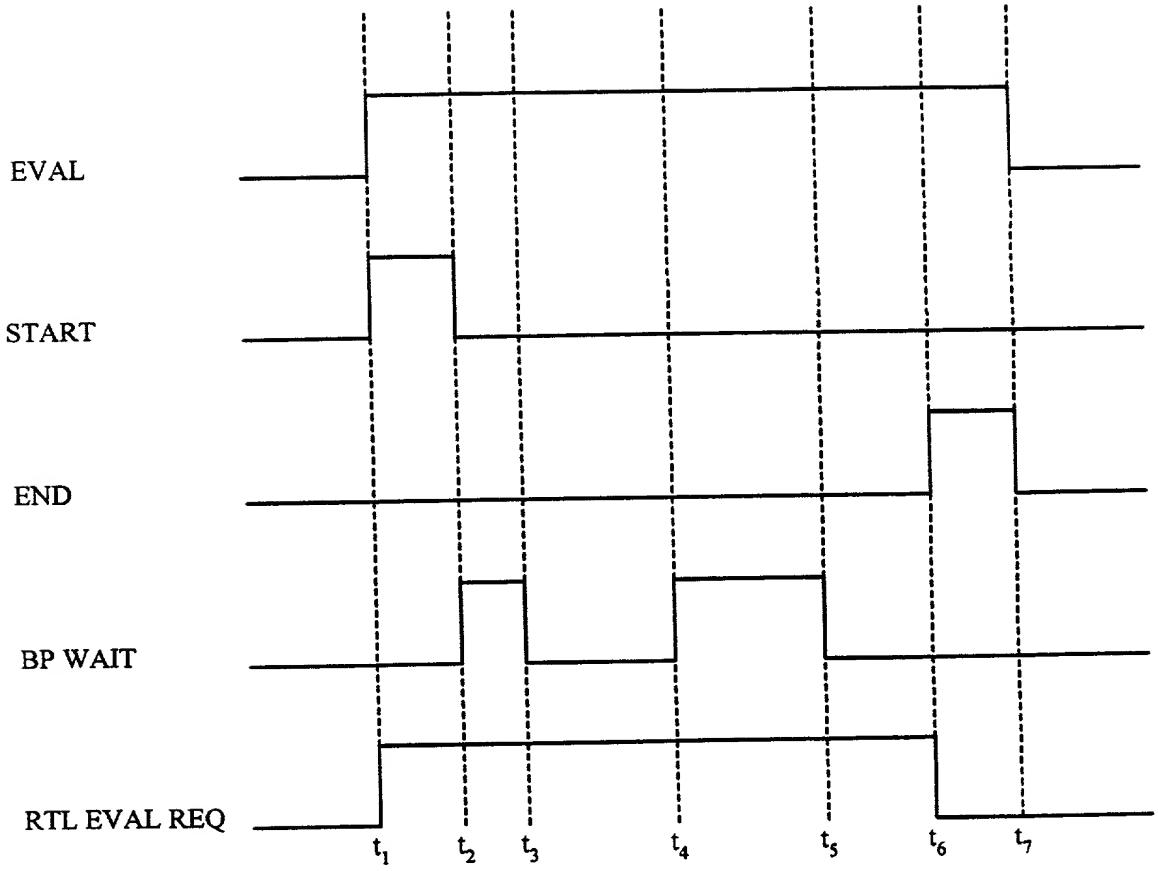


FIG. 104